DOCUMENT RESUME

ED 110 065	IR 002 364
AUTHOR	Plourde, Paul J., Ed.; Thomas, Charles R., Ed.
TITLE	Innovative Systems: Solution or Illusion? Proceedings of the 1974 CAUSE National Conference. Volume 1.
INSTITUTION	College and Univ. Systems Exchange, Boulder, Colo.
PUB DATE	75
NOTE	700p.; For a related document see IR 002 365
EDRS PRICE	MF-$1.23 HC-$34.91 PLUS POSTAGE
DESCRIPTORS	*Administration; College Administration; Colleges; Computer Oriented Programs; *Computer Programs; Conference Reports; Data Bases; Data Collection; *Data Processing; Electronic Data Processing; *Higher Education; Information Systems; Management Information Systems; *Management Systems; Systems Analysis; Systems Analysts; Systems Concepts; Systems Development; Universities; University Administration
IDENTIFIERS	Administrative Data Processing; CAUSE; College and University Systems Exchange

ABSTRACT

Addresses, technical papers, and descriptions of institutional administrative data processing (ADP) applications presented at the 1974 College and University Systems Exchange (CAUSE) National Conference are compiled. Topics include (1) operational data systems, (2) management information systems, (3) planning and management systems, (4) new developments in computer software for instructional and ADP operations in colleges and universities, and (5) specialized ADP applications for higher education. Descriptions of selected proprietary operational data and data base management systems are also included. (DGC)

****
* Documents acquired by ERIC include many informal unpublished *
* materials not available from other sources. ERIC makes every effort *
* to obtain the best copy available. nevertheless, items of marginal *
* reproducibility are often encountered and this affects the quality *
* of the microfiche and hardcopy reproductions ERIC makes available *
* via the ERIC Document Reproduction Service (EDRS). EDRS is not *
* responsible for the quality of the original document. Reproductions *
* supplied by EDRS are the best that can be made from the original. *
****

ED110065

Innovative Systems: Solution or Illusion?

Proceedings of the
1974 CAUSE National Conference
Volume I

December, 1974
Arlington, Virginia

U.S. DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

Edited By
Paul J. Plourde
and
Charles R. Thomas

IR 002 364

CAUSE *College And University Systems Exchange*
737 Twenty-Ninth Street, Boulder, Colorado 80303
(303) 447-9441

2/3

INTRODUCTION

The theme of this year's conference, "INNOVATIVE SYSTEMS: SOLUTION OR ILLUSION?" is symbolic of the increasing controversy of the role of management information systems in higher education administration.

The goal of the Program Committee was to provide a wide-variety of presentations on topics of interest to the administrative data processing community, their managers and users. While the emphasis was on innovative systems, the selection of speakers and topics was made with the recognition that there are alternative approaches to solving the administrative systems dilemma.

The Proceedings of the 1974 CAUSE National Conference reports the efforts of over ninety speakers and nearly four hundred participants to share their ideas, systems and experiences.

The commitment of CAUSE to the concept of sharing extends beyond establishing an annual forum as evidenced by the following statement of the goals of the organization:

1. To serve as a clearinghouse for an active program of exchange of information, application programs, and information systems among institutions of higher education;

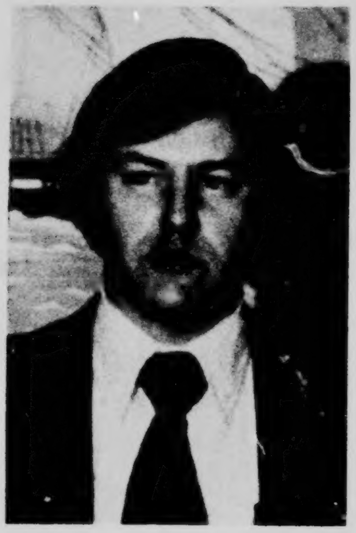
2. To serve as a catalyst and to provide support for cooperative ventures among institutions in the development of improved systems, operations and standards;

3. To promote the general advancement of information systems technology and data processing management in higher education; and

4. To provide a professional association for administrative data processing in higher education.

We encourage the reader to join CAUSE in this information exchange process that will assist each of us in achieving our individual goal; to provide more useful information to support improved institutional management.

Paul J. Plourde
Conference Chairman


Paul J. Plourde
Vice President
CAUSE

Charles R. Thomas
Executive Director
CAUSE

The Proceedings of the 1974 CAUSE National Conference are published in two volumes. Volume I contains the papers from Tracks I through V; Volume II contains the contributed papers from Track VI.

ACKNOWLEDGMENTS

The success of this or any conference relies entirely on the contributions of people and supporting organizations.

The effort involved in such an endeavor is extensive and the voluntary contribution of time and energies by this year's Program Committee was reflected in what would have to be called by any measure an "outstanding conference."

From left to right, First Row: Bob Oglivie, Don Tiernan, John Gwynn, Dick Mann, Ken Shostack. Second Row: Jane Ryland, Betty Dean, Paul Plourde, Cheryl Traver. Note: Bard White was absent at the time the picture was taken.

The continuing support of the CAUSE Board of Directors and the membership whom they represent is gratefully acknowledged.

From left to right, First Row: Jim Morgan, Jack Thornton, Dick Leurig, Mike Schouest; CAUSE Staff Members: Debbie Smith, Chuck Thomas. Second Row: Jane Ryland, Bart Carlson (Secretary/ Treasurer), Bob Taylor (President), Paul Plourde (Vice-President), Cheryl Traver. Note: Frank Martin was absent.

While the Proceedings reflect the contributions of the presenters, the ultimate success relies heavily on the participation of individual attendees and the establishment of a dialogue between these two groups. This year's participants represented a wide spectrum of interests and, whether attendance was motivated by a manager's desire to expand his knowledge or from a technician's point of view, their attendance was the key to this successful enterprise.

Finally, CAUSE especially acknowledges the financial support of the organizations identified in Track V of the Proceedings. In addition, we thank the IBM Corporation for their continued support and contribution which assisted with the publication of this book: INNOVATIVE SYSTEMS: SOLUTION OR ILLUSION?

COLLEGE AND UNIVERSITY SYSTEMS EXCHANGE

NATIONAL CONFERENCE PROCEEDINGS

INNOVATIVE SYSTEMS: SOLUTION OR ILLUSION?

Arlington, Virginia December 9 - 11, 1974

Volume I

TABLE OF CONTENTS

Page

Management and Information in Higher Education
John D. Millett 3

Information Systems for the Congress: Solutions or Problems? (Abstract)
Frank Ryan 15

Management's Greatest Source of Power
Carl Hammer 19

The Power of Wordless Persuasion
W. J. Beeners 33

The Maturing Information System--Social Organization Interface in Higher Education: Accommodation or Isolation?
Marvin W. Peterson 41

The Case for Top Level Involvement in Management Information Systems (Abstract)
Ronald W. Brady 67

Request for Proposals: Financial Systems and Hardware
John F. Chaney and James S. Martin 69

Selecting a New Computer
Don Mittleman 83

Management Appraisals Systemation Before Mechanization
David A. Wearley 109

Management of a Distributed Data Base
B. D. Cross 119

Management Systems in Higher Education (Abstract)
Warren W. Gulko 129

Individual papers in this publication are available, at the cost of reproduction, through the College and University Systems Exchange.

Page

Decision Making in the Early Stages of Developing a New Administrative System (Abstract)
Wayne Muth 131

The Question of a Synthesis in Higher Education Management
Bernard S. Sheehan 133

A Centralized Student Distribution System (Abstract)
Gary Cooley 155

Statewide Management Information (Abstract)
Michael Abbott 157

Structured Programming From a Management Prospective
Robert L. Mason 159

Personnel Considerations in Project Management
Charles G. Foskey 173

Inquiring Systems for Higher Education: A Perspective Based on Theories of Development of Organizations, Individuals and Decision Structures
George Weathersby 197

Software Products: What's Available the Benefits, and the Pitfalls (Abstract)
Richard Clikeman 221

Implementing a Library Book Ordering System at the University of New Hampshire Library
Conrad C. Born 223

Teleprocessing Case Study an On-Line General Stores System
Al J. Dandrea and M. M. Bruneau 233

Implementing an On-Line Data Base System for Personnel Records
Pat Urban 249

The Successful Implementation of an Automated Student Loan Accounting System at Louisiana State University
Michael E. Schouest, J. Glenn Gremillion,
John R. Davidson, Leslie R. Lavergne 277

An Overview of Interpretive Structural Modeling
David W. Malone 297

Telecommunications System Case Study: Student Registration and Scheduling (Crossing Over: Batch to On-Line)
Joseph A. Duerstock 307

Page

Fundamental Benefit-Cost Consideration for System Design: The MIS Information/Activity Matrix
Robert Harris 323

Designing a Batch System for Transformation to a Teleprocessing System
Judith Campbell and J. Paul Wetterau 331

Physical Plant Operations Cost Management
Don Hennings 351

Physical Plant Maintenance Scheduling (Abstract)
R. Wayne Ingersoll, Jr. 365

Feature Analysis: Personnel-Payroll Systems
David P. Tursan 367

Use of Act Data at Kansas State University
Mark E. Lipp 373

The Role of the Consultant in Systems Design (Abstract)
Peter Simmons 383

Why Data Base?
Tom Richley 387

A Data Base Management System Approach
Adele Mildred Koss 397

A More Conventional Systems Approach
Eugene E. Ekstrand 411

Planning for a Database
John K. Lyon 419

Procedure for Establishing Data Standards at Cornell
Lynne K. Personius 439

Using the ADABAS System
David Benevy 455

Data Dictionary Systems
A. H. Cassidy 465

Data Base Administration, IMS, and the Master Data Base
William B. McKelvey 479

Page

Summary of System 2000 Data Base Management System Evaluation Project
Mark E. Ragel and Martha A. Fields 489

The Data Dictionary/Directory: Your Most Important Data Base
Daniel S. Smith 511

MARK IV With IMS
Vinod Chachra and A. Wayne Donald 523

Data Base Development in a Multi-Campus University
Richard L. Mann and Dorothy J. Hopkin 539

An Evolution in Administrative Data Processing
Joanne M. Gallitano 561

The Trend Toward A Common Data Base: Languages and Structures
Mike O'Connell 589

Directions in System Development: Key to Enhanced Productivity (Abstract)
Gale R. Aguilar 595

Future Computerized Educational Systems (Abstract)
Earl C. Joseph 597

Implementing Structured Programming
Donald H. Bein 599

The Realities of the Used Computer and Third-Party Lease Market (Abstract)
George Spanos 609

Distributed Processing Systems (Abstract)
David J. Farber 611

Reality, A Micro-Programmed Data Management System (Abstract)
Richard Pick 613

Mini-Maxi Mixing: A Hierarchy for Academic Computing
John H. Esbin 615

The Development of the Massachusetts State College Computer Network
Leo Roomets 649

Planning, Installing, and Using Remote Batch Terminals
G. E. Bird 661

Page

When are Microfilm Systems Better Than On-Line Systems: Can COM Compete?
Alex Wenger 669

Computer System Performance, Measurement, and Analysis
James C. Browne 675

Office Automation and Hybrid Information Systems
Jack E. Shemer 689

Student Loan Systems (Abstract)
Dan R. Ackley 697

IDMS - Integrated Database Management System CULPRIT Information Retrieval System (Abstract)
William Casey 699

MARK IV - File Management System
Al Seyler 701

System 2000 as an Administrative and Scholastic Aid (Abstract)
Charles J. Lewis 703

ADABAS - Adaptable Data Base System (Abstract)
John N. Maguire 705

The Aquinas Principle and its Application to the Implementation of Data Base Payroll Personnel System (Abstract)
David A. Duffield and Michael P. Aherne 707

How to Make the Best Use of Consulting Companies (Abstract)
John Robinson 709

Inquiry Systems - Query 5 (Abstract)
David McIlhenny 711

Interactive Query and Report Processor (IQRP) (Abstract)
J. F. Movizzo 713

Microfilm and the Computer: Best of Both Worlds (Abstract)
James Rasnick 715

How to be a Hero Every Day of DP Life (Abstract)
Robert Briggs 717

Implementing SCT's Non-Package Locus Within Both Batch and DBMS Environments (Abstract)
Jack F. Magosin and John Baillis 719

Page

TOTAL - A Successful Approach to Data Base Requirements in a University Environment (Abstract)
Michael J. Ehrensberger 721

Achieving a Passing Grade in Data Base (Abstract)
Leo J. Cohen 723

WISE - Administrative Data Base Management Along With Instructional Time-Sharing (Abstract)
Robert H. Meese 725

KEYNOTERS

Frank Ryan

John D. Millett

MANAGEMENT AND INFORMATION IN HIGHER EDUCATION

John D. Millett
Vice President
Academy for Educational Development

Management and Information
in
Higher Education

by

John D. Millett
Vice President
Academy for Educational Development

Management of our instituions of higher education has been in a continuous state of crisis for at least 50 years. It has been almost that long since I became a high school student preparing for college entrance, and I can't recall any time in all the intervening years when there has not been some evidence of stress and turmoil on our college and university campuses. Perhaps we should admit that higher education by its very nature is always in disarray.

Management in our colleges and universities is in trouble for several reasons. In the first place, just as we always insist that higher education is a unique social institution among the many institutions of a pluralistic society and of a liberal democracy, so management of a college or university is unique. This management endeavor is different in fundamental ways, I am convinced, from management in business, in government, in a religious denomination, in the military, or in other institutions. In the second place, management in higher education must seek to encourage faculty members and students to accomplish goals that at best remain both uncertain and illusive, and in the third place, management in higher education must obtain resources for the enterprise from a variety of sources, some or all of which from time to time express reservations about the utility of the entire endeavor.

I wish there were time and space here to discuss these issues I have raised in the detail their importance deserves. I can do no more here than call attention to one of the studies sponsored by the Carnegie Commission on Higher Education, the study by Michael D. Cohen and James D. March, published in 1974, and entitled Leadership and Ambiguity. Cohen and March describe a college or university as an "organized anarchy", and they ascribe the "general properties" of this organized anarchy as problematic goals, unclear technology, and fluid participation.

I find very few college administrators or faculty members who are particularly happy about either the Cohen and March label or the description that follows. And yet I am personally convinced from my own experience and observation that Cohen and March have provided us with an important insight into higher education as a social institution and into colleges and universities as particular enterprises. We must still in this country come to grips with the meaning and mechanisms of management within the context of an organized anarchy.

Our interest in this conference is with the uses of information in the management of higher education. At the same time we must not be unaware of or unconcerned with the limitations of management as a structure and process within a college or university. At best, management in higher education seeks to encourage and promote an environment of learning. Management seeks to achieve this environment within the context of an enterprise that does indeed have imprecise goals, an unclear technology, and uncertain external support.

We are all aware that our own era is one given various kinds of labels: the age of discontinuity, the age of technocracy, the age of confrontation. Among other designations we are told that ours is an age of knowledge, and that the knowledge industry has become one of the major economic endeavors of our time. For the moment, I suspect, that the energy industry has replaced the knowledge industry as a matter of concern to the American people, and yet the knowledge industry is a definite characteristic of our economy.

The knowledge industry as I see it really consists of two somewhat different if interrelated parts. One part is the industry concerned with the production and communication of knowledge: our colleges and universities, our professions, our research institutes, our book-publishing, our museums, our libraries, our media of mass communication to the extent that these can be separated from entertainment, and our cultural agencies. The other part of the knowledge industry is that involved in the processing of information. Thanks to electronic data processing there is not an enterprise or a government agency today that cannot literally have at its fingertips a vast array of factual and analytical data.

Indeed, the first observation I want to make about higher education management and information is just this. There is a very real danger that the availability of information may become an end in itself, and especially within higher education. Those of us concerned with colleges and universities are likely to be of a frame of mind that holds any and all information to be useful. But from a management point of view, information for the sake of information is a wasteful expenditure of resources. Management in a college or university needs information useful for management.

To be sure, government agencies are expecting an ever increasing flow of information about our colleges and universities. No one apparently knows exactly how much additional administrative cost has been thrust upon colleges and universities by the demands for information about personnel practices, contracting practices, financial practices, enrollment practices, and a variety of other procedures. Many of us are aware of how the rise in energy prices has affected and will further affect the costs of college and university operation. We need to be equally sensitive to the costs of information processing.

This year one executive officer of a state government board of higher education vigorously protested to a federal government agency about the costs of providing the information that agency wanted about student enrollment. The protest was more than justified, and for once had some effect. Insofar as I know, that particular request for information has not yet been officially demanded.

I know of one university president this past year who instructed his staff not to respond to any questionnaires unless the request for information was accompanied by a reimbursement of cost for time and energy consumed in responding. But this is a policy that colleges and universities cannot adopt in their relationship to state and federal government agencies. Ths institutions that accept state and federal funds pay the price of responding to state and federal requests for information.

In two or three state governments recently I have come across the situation where state agencies have requested a substantial amount of data from colleges and universities within the state. The institutions responded, and the data provided were useless because of their very magnitude. And I suspect the same kind of circumstance can arise within a college or university itself. So much information can be generated that its very volume defies use.

Colleges and universities, and the governmental agencies concerned with their operations, must be alert to the uses of information. Even more, I would say that the only information collecting and reporting that can be justified is the information for which there is a definite use and for which there is a reasonable cost. Information for the sake of information is simply not a justification. Information essential for management is the only reason to obtain data.

Some definition of both the nature of higher education management and the information needs of higher education management would be appropriate here. These matters again require more extensive consideration than is possible here. We can do no more than acknowledge the fact that colleges and universities are expected today to have essential data available about inputs, process, outputs, and costs. The primary inputs are enrollment, faculty, learning support, and institutional support. The process is a technology of learning, consisting of courses, degree requirements, and faculty work processes. The outputs are student instruction, knowledge, creative activity, and service. And the costs are the expenditures involved in the entire system.

When I was Chancellor of a state system I constructed our information system around four subject matter areas: enrollment and student characteristics, staffing and faculty activity, space inventory and utilization, and income and expenditures. All of this information was essential to the state government role in providing financial support on an equitable basis to the various state sponsored institutions of higher education. The operative word here of course is that of "essential".

The second observation I wish to make has to do with the importance of an analytical framework for the use of knowledge. There is only one way in which a considerable input of data can be made useful and employable for management purposes. That way is to construct in advance an analytical procedure whereby a mass of data are consolidated and analyzed in terms of a few essential facts. Most of the failure I have observed in the handling of information requests is a failure to decide in advance how to analyze the desired data and how to make use of the data when available.

Electronic data processing is capable of storing and of providing large amounts, even masses, of information. But information by its very bulk may defy the human capacity to understand it and to use it. Most of us can comprehend only a few related facts or figures brought together in a fairly simple matrix or table. The task of analysis is to produce that matrix or that table the administrator can use. In the absence of an analytic framework, data processing has little if any utility for decision-makers.

To be sure, one of the characteristics of our day is the expectation that administrators will know, will understand, and will use essential information about the enterprise for which they are responsible. It is not enough for decision-makers to have a philosophy of higher education, or an intimate awareness of institutional history. These kinds of background information are important. But commitments and perspective are not a substitute for current facts. Those who provide support for college and universities expect presidents to know the enrollment and the program interests of students, the number and interests of faculty members, the number of professional and other support personnel, the extent of facility resources and their use, the costs of instruction by major programs, other output costs, the costs of student aid, the costs of overhead, and the sources and use of available income.

Private foundations, alumni, government officials, legislators, and others expect hard data, not impressionistic platitudes from higher education administrators. As higher education became affluent, it lost autonomy as well as the bliss of simplicity. Responsible, accountable management is an informed management, knowing and using essential data about the enterprise.

Furthermore, higher education administrators must have information to share with faculty members, students, and staff. One of the unfortunate failings of some administrators is a reluctance to share bad news with others. I once knew a very prominent executive in the business world who told me that there were only a few persons in any organization who had the personal characteristics needed to be planners. And the most important needed characteristic was not to be afraid of the future, not to be upset by the projections of future magnitudes.

I suspect that a great many persons in any enterprise, indeed in our whole society, are afraid of the future. Most projections of things to come are frightening rather than reassuring. But in our colleges and universities faculties and students have insisted that they wish to participate in the decision-making about their future. So be it. If faculties and students wish to participate in decision-making, then administrators must share information, all available information, with them.

I have been on one university campus recently where faculty members and student members of the university senate told me that administrators had responded to every request for data they had made. These persons reported that they drew somewhat different conclusions from the data than did the administrative staff, but at least they had all shared the same data base in their discussions about the future of the university. This kind of circumstance is the one that ought to exist on every campus.

On the other hand, I have visited on another campus where some faculty members and some students believe that too much information was shared. The perception here was that the president of the university had done more than share information; in the process of sharing data the president had appeared to abandon a role of leadership. As a consequence, the university felt that the dire events it had experienced somehow were related to too much information sharing. I would be disposed to say that the fault, if any, should be ascribed to a want of firm leadership rather than to a surfeit of data.

I recall an old story I heard as a youth. It went something like this: that the basic difficulty in almost every situation was too much or too little of some desired good. I think this diagnosis applies to most information gathering, analysis, and dissemination. There is either too much or too little information.

A third and final observation has to do with this matter of information as the basis for anticipating the future. The larger the enterprise in which administrators function, the more restricted is the administrative role. In a large enterprise with many and different operating parts, the authority and responsibility of current operations must necessarily remain with the parts. The alternative to this practice is apoplexy at the center and paralysis at the extremities. There is probably no characteristic more important to a high level administrator in a complex and sizeable enterprise than the ability to delegate authority, and the willingness to accept responsibility for results.

The competent administrator in the large-scale organization exercises his authority by his capacity to anticipate and plan for the future. And the administrator who wishes to be effective in planning must be one who can handle information about the present and the past. No one is necessarily a prisoner of the past, and no one must necessarily expect the future to be a linear extension of recent experience. But the difference between a dreamer and a planner is the difference in information analysis and assessment. The dreamer ignores limitations of resources and technology. The planner cannot afford to ignore these limitations; he must ask whether or not to modify goals and aspirations, whether or not new resources can be found or developed, whether or not new techniques of producing desired objectives can be realized.

I would emphasize that management is above all else planning, is preparation for the future. This preparation is the major task of management in higher education. But this management task can only be accomplished from an adequate information base. The planner in higher education begins with experience and then endeavors to influence the future by actions that adjust the parameters of the past.

Planning for the future is the largest challenge facing higher education today. It is a challenge that can only be met with the courage to innovate, the will to influence events rather than to surrender to them. But the courage to innovate and the will to change have some hope of achievement only if information analysis and assessment have preceded action. Such analysis and assessment are no guarantee of success. But the absence of such analysis and assessment are almost a certain prediction of failure, except for the most fortunate men and women of society.

Higher education management needs information. And then higher education management needs the capacity to know how to use information as the basis for trying to achieve a desirable tomorrow. Just as human intelligence is our product, so also is human intelligence our only hope for the future of higher education itself.

INFORMATION SYSTEMS FOR THE CONGRESS: SOLUTIONS OR PROBLEMS?

Dr. Frank Ryan
Director
House Information Systems
Congress of the United States

INFORMATION SYSTEMS FOR THE CONGRESS: SOLUTIONS OR PROBLEMS?

Dr. Ryan focused on the application of information systems technology in the congressional environment and examined the problems arising from this application and solutions found to it. He discussed the political realities which added yet another dimension to systems development; one in which a constant battle is waged between effective systems and effective appearances.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

KEYNOTERS

Carl Hammer

MANAGEMENT'S GREATEST SOURCE OF POWER

Dr. Carl Hammer
Director, Computer Sciences
UNIVAC

Management's Greatest Source of Power

January 1975

Carl Hammer, Ph. D.
Director, Computer Sciences
UNIVAC, Washington, D.C.

Society is a vast and complex accumulation of living organisms and inanimate objects in a thermodynamic environment, a dynamic system in the most general sense. Subtle interrelationships exist in time and space between its component subsystems: people, families, cities, states, and organizations all of which are motivated and stimulated by many endogenous and exogenous forces. For man to exist and prosper in his ecology, he must adapt and respond intelligently to all inputs received from his environment. He must provide food and shelter for himself; he must also accept and process data or information, make rational and intelligent decisions about optimal management of his limited resources, and transmit meaningful signals to others around him.

Our global society must be looked at as a whole, as the synergistic sum of its components and not just the parts by themselves. We must employ the concepts of communications and control theory to interpret society as a purposive system. Its subsystems are no longer loosely coupled; their interactions have deepened in scope. Society tends simultaneously toward randomness (increased entropy) and organization (decreased entropy). Powerful sophisticated models of our cultural, political, and economical environment are being developed on the basis of far-reaching assumptions about the ontological or teleological nature of societal systems.

As computing power engulfs this world we take the "final" step toward a mind-amplifying, information-rich society. As we understand it today, this will be a major milestone in our evolutionary process. It presages a new orientation with perhaps devastating economic and political consequences. During the middle of the last century management still believed that the human work force was its most important asset. At the same time organized labor began to flex its muscle, asking for a greater role in the management process -- and for higher wages paid for doing the menial jobs in factories and industries. The philosophies of Karl Heinrich Marx (1818-1883) and Fredrich Engels (1820-1895) were derived from this viewpoint which was certainly correct at that time. However, with the beginning of this century industry became capital-intensive. As the pervasive force of automation grew, management noted that machines had become their most important asset! The capital investment for new machinery became, in fact, a yardstick for the more affluent, industralized nations. Meanwhile, the political structure of several emerging societies clung stubbornly to the doctrines of Marx and Engels, allowing little room for management advances and "revisionary" thinking. As we continue our own evolution, these societies still refuse to break openly with their dated tenets.

Today, neither labor nor machines are management's greatest source of power: Electronic systems have moved possession and control of data and information into a position of supremacy. We can always hire more workers in the open labor market; and we can also always buy more machines to manufacture more goods for consumption. But if we lose our data base from which management plans and operates, we face chaos and/or ruin! The story has been told innumerable times, of the manufacturer who "lost" his Accounts Receivables Tape because a disgruntled programmer decided to "get even" with his employers. As a consequence his company went bankrupt, being unable to reconstruct the lost data base. The information stored in our computers is considerably more vulnerable than the machines which the Luddites attacked at the beginning of the last century. Thus computer and data security are now the focus of management's attention -- data and information have become our most valuable assets!

Electronic data processing systems have created many new problems such as the need to develop more effective interfaces with their human "masters". The trend has been toward reduction of human intervention in purely operational matters and the transition to real-time systems which can be more responsive to management's immediate needs. The design and development of management information systems has made great progress. They begin to deliver more timely, accurate, and pertinent information at a reasonable cost. As a result, we can now make better or faster decisions and we can allocate our limited resources more effectively.

Advanced hardware has become available for displays and intelligent terminals; but the resultant synergism of computers and communications has again created new challenges and problems. Static management systems of the past are being converted to make them dynamic and interactive. Human decision making processes are being examined for consistency and algorithmic properties; perhaps we can codify them and harness the decision making power and artificial intelligence from computerized models. Graphics and other displays in real-time are beginning to displace classical ledgers and even computer printouts; managers can thus direct their talents to the recognition of patterns and the systematic examination of causal or hierarchical relationships.

These advances have not come about without a struggle. Managers simply do not like to be cast in the role of pseudo-technologists. But in order to keep up with the rapid pace of the new technology, they must pay more than lip-service to increasing their computer-oriented skills. Experience shows that successful systems are most often designed in-house, not by outsiders or consultants. It also shows that you should "never bite off more than you can chew" when acquiring new hardware or implementing new applications. Generalized software systems are very complex and truly needs-responsive packages are therefore often developed in-house. Before ordering a score of displays or terminals, the wise manager starts with one each, to learn how to plan for expansion. He does not introduce the more sophisticated devices, such as Lightpens, Rand Tablets, Mouses, or Joy-Sticks, before experimenting with a simple keyboard/printer in a truly interactive mode.

Nevertheless, successful applications of information technology abound. With an estimated twenty thousand displays now serving bus, rail, and airline reservations in real-time, we would be hard pressed to operate our complex transportation systems without them. Engineering design is beginning to make use of graphic systems with lightpens; educational systems are experimenting with computer-aided, even computer-managed instruction. Management planning and control have taken on new dimensions with our ability to select, in real-time, current data and summaries from centralized or distributed, tailor-made data bases. More advanced systems with automatic controls are coming into operational use. While the cost of hardware is decreasing steadily, the cost of developing and implementing large systems seems to be on the rise. Five key points to be remembered in the planning and selection of information management systems are:

(1) Establish well-defined objectives and allow sufficient time for systems design, development, and implementation.

(2) Consider trade-offs in the development of management information systems software and use in-house talent only if it is available and competent.

(3) Purchase packaged applications even if they don't quite fit the stated objectives. It is often better to force-fit the objectives than to flounder in the rough sea of writing special-purpose software.

(4) Start real-time interactive systems with simple terminals and add capability as experience and requirements grow.

(5) If the present system works, don't tamper with it; the glamour of pioneering more advanced systems has a high price tag.

Advanced information systems for management have received much attention in the literature. Many published studies indicate that their goals are not as readily achieved as would be apparent from the large number now in existence or planning. It has been pointed out that the problem of organizing an effective management information system becomes progressively more complex as computer applications are extended in scope. Present concepts seem to have fallen short, in many cases, of providing a real solution to the problem, despite the guise of "total systems", "integrated systems", "real-time systems", or other clichés with which the field abounds. Many such systems provide, at best, dated and voluminous outputs which tell little more than could have come from "ye olde tab shoppe".

Nevertheless, a good case can be made for electronic data management systems. It is safe to predict that future managers will indeed be able to make intelligent use of electronic intelligence systems, to paraphrase a well-known expression by Norbert Wiener. Some day such systems will provide not only factual data in real-time, but also statistical extrapolations, decision functions, and even the calculated risks associated with alternative courses of action. If the snail's pace of these developments troubles us we should remember that history took millenia for the development of societal and governmental systems. With the current population explosion, society grew both in terms of sheer number and structural complexity. There is a saying among technologists that complexity begets complexity; the more complex our systems become, the more complexity they demand in return.

More than once, Margaret Mead has pointed out that this may be a major flaw in the evolution of societal systems. The race between their growth in complexity and concomitant demands on advances in computer technology cannot continue unchecked forever. We are either headed for a catastrophic breakdown of "the system" or we decide collectively to reduce the systems complexity level. Cyberneticists all over the world wonder indeed whether mankind will have the courage and wisdom to declare a moratorium on complex systems -- or become slaves to "Big Brother" in 1984. However, as long as lawmakers fail to heed the warning signals, we see little hope to bring about such an achievement on a voluntary or planned basis. The prevailing attitude seems to be that technology can always cope with one more problem and no one surmises that we are perhaps rapidly approaching the proverbial enactment of the "last law which breaks the computer's back".

Technology and society are, in fact, totally integrated but the complexity is such that few can comprehend its total dimension. Yet there exists one common component: Data Trails. How many data were left behind by the aborigines who inhabited the jungles ten thousand years ago? Zero bits per person per lifetime: They were born, led a miserable life trying to survive, and most people died at an early age. But we don't have as little as a headcount of them!

And how many data do we leave behind today, in brown folders, on punched cards, or magnetic tapes and discs? On the average, one million characters of information per person per year are thus recorded. To accommodate this avalanche, data banks are springing up everywhere. In fact, even their number is not known exactly! Like olives in martinis, they come in all sizes: large, extra large, giant and colossal. The rapid growth of the computer industry is perhaps a sad tribute to our mania for collecting, processing and preserving data on such a scale.

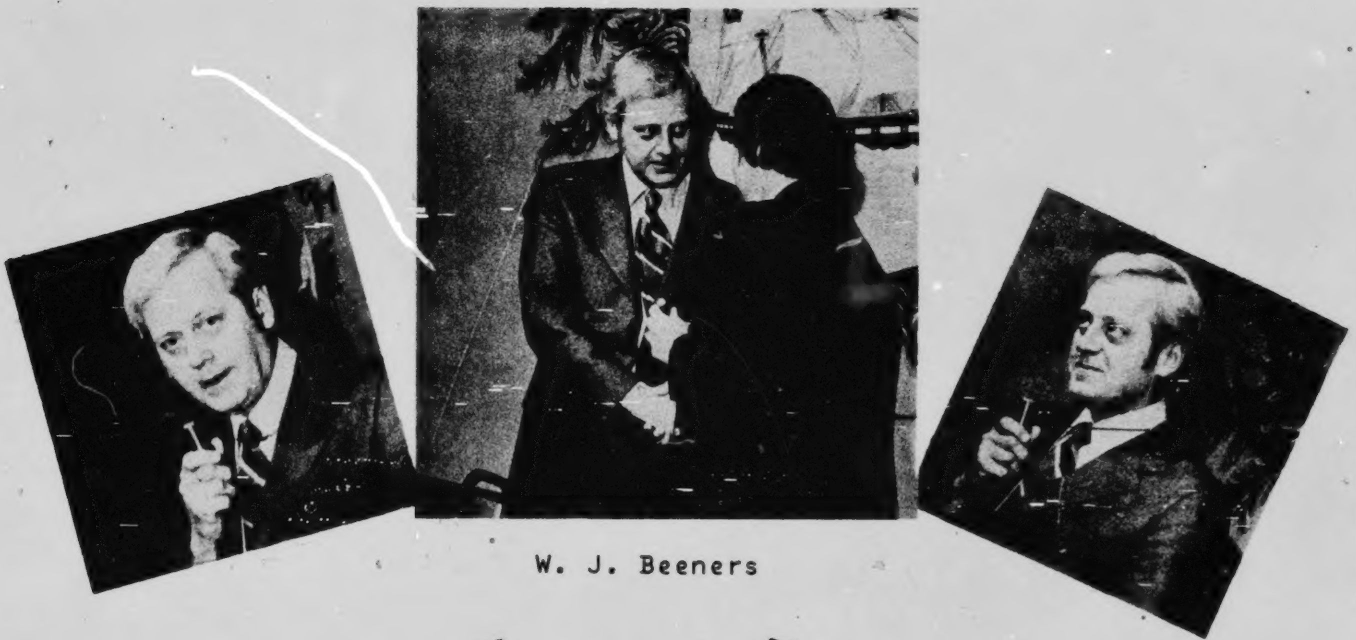
There is no question that the U. S. is the world leader in developing this new form of power, in building machines to absorb activities formerly done by man himself but now relegated to the computer -- as they properly should be. Jean-Jacques Servan-Schreiber, the well-known French journalist and politician, examined this development in "The American Challenge". He asserts that we achieved our position of economic dominance because we are foremost with computers. He also points out that electronic systems differ fundamentally from the earlier muscle-amplifying machines; they function as mind-amplifiers!

To date, well over 100,000 computer systems are installed in the U.S. Some of these are super-systems which can outperform millions of human beings. It must be shocking to realize that our aggregate data processing capability equals a clerical workforce of 400 billion humans. This enormous electronic mind-amplifying power supports our integrated economic-political system where better than 99 percent of all clerical work is now done by machines and the percentage keeps growing.

As we ponder this proliferation of computing power we wonder perhaps why we need all these machines in the first place. The answer stated in the most simplistic terms is rather revealing: Because societies are fundamentally corrupt and mankind is basically dishonest! Computer toil forever with payroll, inventory, data base management and untold other accounting type applications; they also keep track of crime and criminals. But people are always stealing, lying, and cheating -- forever violating all basic rules of ethical conduct! If we lived in a perfectly honest society, we would need computers only for scientific, statistical, or humanitarian purposes. What would such a society look like? For an example, consider again the payroll application. A messenger leaves his office each Friday morning and goes to the nearest bank, where all the cash is kept in an open vault -- to protect it from the rain, of course! He scoops up the needed money, puts it in a basket, returns to his office, and deposits it in the building entrance. As the employees leave on Friday afternoon, they take out their own, honest share -- even make change. We can do it with newspapers but not with larger transactions! As long as man is basically corrupt we will need computers, and more computers, to do battle against "the forces of evil"!

In the future, electronic systems are bound to be even more powerful, faster, and reliable. They will operate in a real-time mode for practically all applications, providing computing power as the utility companies of today provide electric power. But the most spectacular advance is likely to be the introduction of millions of low-cost terminals into our homes, allowing many jobs to be done without travelling to an office. Thus electronic systems are destined to take on practically all the tasks of rote and drudgery which nature and society now impose upon us. This event was forecast long ago by Norbert Wiener when he observed that "It is a degradation to a human being to chain him to an oar and use him as a source of power; but it is an equal degradation to assign him a purely repetitive task in a factory (or office) which demands less than a millionth of his brain capacity." In the end, electronic systems will benefit all of mankind by allowing society to achieve its ultimate goal: Make human use of human beings!

BANQUET



W. J. Beeners

THE POWER OF WORDLESS PERSUASION

W. J. Beeners
Director of Speech
Princeton Theological Seminary

The Power of Wordless Persuasion

The word "communication" is perhaps the most often used and abused in the English language. It has become a convenient scapegoat for many of our problems on the domestic as well as international scene.

While we use the word with facility, when it comes to inter-personal relations, apparently we are not very good at communicating. If we were any better at it there would be fewer divorces, separations, suicides, crimes . . . perhaps less misunderstanding at the office. As theologian Roger Schinn observed, "We are living in a world saturated with communication, on the verge of perishing for lack of it; a world smothered with words, hungry for one meaningful word; a world bombarded with data, rarely capable of sorting out the truth; and a world in which we can flash messages across the ocean by way of space, but one in which we find it difficult to get through to each other face to face."

The reasons for such failure are many and complex. Sometimes we do not accurately say what we intend. How often has someone said to you, "But, sir, what do you really mean?" How often have you said to your spouse, "But, honey, where did you get that idea? I didn't say that at all." Perhaps you've noticed the plaque, "You say you understand what you think I said. What you don't realize is that what you heard is not what I meant."

It is true also that we have been so bent on expressing our own ideas that we have neglected the person on the receiving end. Much that has been criticized as poor communication has been excellent expression. As one man put it, "Just because you sent the message,'t'aint no sign he got it." Communication is a complex juggling act involving the sending of many signals between two or more persons. It takes sensitive listening.

Perhaps the chief reason for our failure to communicate is that we actually do not want to. It is a threatening engagement. The listener often fears manipulation when spoken to directly. Remember, the patient is more persuaded by what he overhears than by what he is told.

And the initiator of communication fears not only rejection but also self-revelation. We reveal ourselves whether we like it or not. Sigmund Freud stated it well, "No one can keep a secret. If a man's lips are silent, he will be chattering a message with his fingertips. Communication oozes at every pore."

Recent studies have discovered much about the totality of personal communication. Words, for instance, only rate approximately 10% reliability; tone and body movement 90%.

Words are excellent tools for the dispensing of ideas and the giving of information. But when it comes to expressing how we feel about those ideas, tone and action say it for us. What else is sarcasm but the contradiction in manner of what we say in word? We have long ago learned that people are more moved by the depth of our convictions than by the logic of our ideas. Our convictions and values come through our tone and action.

Why not hold a mirror up to the nature of our expression and ask, "Is the impression I am making, the impression I intend to make; and if not, why not; and what can I do about it?"

Most of us know what the body is made up of and how it acts, but how many of you have ever stopped to think of what it says?

The wide base of defiance, the shrunk center of fright, the cocked head of disbelief, the raised brow of indifference, the busy fingers of impatience - are all signals used every day and we need to be sensitive to them.

The tight voiced intensity, the breathy anxiety, the nasal stupidity, the stridently nasal twang, the throaty pomposity - all these speak of feelings and purpose. The inflections of tone share an attitude, often without our awareness - a whine, a surprise, a suspicion, an invitation to argument. Voices may irritate and comfort with equal facility.

When manner and words are in contradiction, what do you believe? Of course, the manner. But when both manner and words are accurately expressing the thought, you are part way, at least, toward effective communication.

COFFEE AND SOCIAL HOURS

As all of you are undoubtedly aware much of the productive work and learning experiences at conferences occur during coffee and social hours. The following candid shots are included as testimony to that observation.

TRACK I

Organization and Management

Bard White and Robert Ogilvie

Marvin W. Peterson

Ronald W. Brady

George Weathersby

Track Coordinator: Bard White

THE MATURING INFORMATION SYSTEM--SOCIAL ORGANIZATION INTERFACE IN HIGHER EDUCATION: ACCOMMODATION OR ISOLATION?

Marvin W. Peterson

Associate Professor of Higher Education
Center for the Study of Higher Education
University of Michigan

INTRODUCTION AND FOCUS

In the beginning, perhaps as far back as 1965, there was higher education. There were students, faculty, administrators, and legislators. They saw it, and it was good.

But on the second day there were continually increasing enrollments and later campus disruptions and spiraling costs.

On the third day the word became "accountability." Legislators came to believe in it. Administrators sought solace in it. They confessed their guilt and turned to planning and management systems, MBO, program budgeting, and other such techniques for their absolution.

On the fourth day technical experts in the guise of systems analysts, operations researchers, computer experts, and management scientists descended in multitudes to assist administrators and legislators. They analyzed and designed information, management, and accountability systems and developed related hardware and software applications for higher education.

On the fifth day, amid this great cacophony of activity, administrators, legislators, and many of the new technical experts extolled the "great awakening." It emerged in its earthly forms; NCHEMS with its many advisory bodies and task forces, many state or federal groups, and endless consulting firms. They all generated "products" for the multitudes.

And on the sixth day there were many implementation attempts of these externally generated products combined with institutional development of new applications. There were increasing disagreements, splinter groups, and many individualistic efforts among the technical experts. The movement grew but with substantial confusion and chaos, especially on individual campuses where academicians and administrators were struggling with new management concepts, computer and information system products, and their newly arrived technical experts.

But on the seventh day many of them discovered CAUSE. Imbued with a new spirit of cooperation and comforted by this new mechanism of exchange, they enjoyed their camaraderie, and they rested.

Such was the first epoch of widescale development of information and management systems in higher education, and perhaps such is today. The first purpose in this paper concerns the issue raised by this epoch. It explores how the rapid development of information systems, particularly the introduction of their new management related approaches, and all their concomitant technologies and staff, induced strains with the many formal and informal patterns of administrative and academic activity within our colleges and universities which needed to be recognized. A little over two years ago in a paper entitled, "Some Organizational Implications of a Management Information System," I highlighted some of those concerns and want to review them briefly with you today. I was concerned that failure to recognize those strains might threaten the successful implementation of improved information and management systems if they were not reckoned with. I would like to review them briefly with you although I think they are strains that you have all experienced and which are being accommodated in your various institutions. But, as with all allegories, we are concerned not only with the hidden meaning of the first epoch, but also with what is emerging and what lies ahead. Thus, the second and more basic purpose today is to move beyond this current struggle to introduce information processing and management systems and to look at several "emerging conditions in our current higher or postsecondary educational environment that could serve to further isolate or to better integrate those involved with the development of information and/or computer systems (particularly the management applications) from the academic and administrative structure of the institution unless they are recognized and dealt with in the days ahead." In discussing these emerging conditions, I will also attempt to identify some of their organizational effects and implications, which raise some questions or concerns about how they might be dealt with.

Throughout my comments there are two major assumptions. First, when I speak of information systems, I am alluding broadly to the process which includes all of those persons or offices in a college or university, often under different administrative structures, who are involved in the analysis and design of information systems and data files, in the design and administration of hardware and software systems for handling data, in institutional data collection, and in the various analytic offices that regularly prepare reports, studies, or analyses. My second assumption is that it is desirable for those who have principal responsibilities for an institution's data system development, the supportive hardware and software, and the analytic staff to be integrated into the academic and administrative decision-making structure of the institution. That is, they need to have close working relationships with both the non-academic administrative staff and the faculty or academic administration and to have their work accepted as a legitimate institutional activity if it is really to have an impact on both improved institutional management and long-term educational effectiveness.

ISSUES PAST AND PRESENT

During the late 1960's and perhaps up through 1972 (the first week in our allegory), the major development of information systems was spurred largely by off-campus forces and groups. It was primarily an activity engaged in by the people with information systems expertise, by non-academic administrators in colleges and universities, and by some staff from state or federal systems or legislators. It was clearly management oriented. It proceeded primarily as an external development of information systems ideas and notions without extensive involvement of other on-campus academic administrators or educational personnel;

that is, many of the information systems ideas and analytic procedures, and the hardware and software applications were developed either external to the campus or in relative isolation on campuses by a limited number of non-academic administrators and information and computer experts who were primarily concerned about the issues of management, accountability, and so on. Further, they were developing information and analytic systems that were built on rational, hierarchical or highly structured assumptions about both decision-making and management procedures and the educational process. They presumed these developments would be non-threatening, and they they would be readily used by decision-makers to improve higher education. The focus was on information system development and not strategies of introduction or implementation. This perspective was largely reflected at the first National Forum on New Planning and Management Practices held in Denver in January, 1972. The program itself was a showcase of new information-based analytic systems and computer hardware and software systems or applications that were being developed. The attendees tended to be either persons directly involved with such developments or administrators with a strong background and inclination toward the business administrative side of college and university administration. In one of the keynote speeches Lyman Glenny highlighted his concern about the new "technocrats" in higher education.

Formal Organizational Strains:

Briefly, the concerns that I had about this predominantly external and rational approach to extensive restructuring of an institution's information and management technologies were summarized in terms of their potential impacts on certain formal organizational characteristics of colleges and universities, certain informal patterns, and their external relations. These information systems developments could potentially collide with four rather crucial formal organizational dimensions of colleges and universities: goals, formal structure, the nature of decisions, and measures of effectiveness.

Goal Diffuseness. Many of the approaches surrounding information systems developments for management uses assume clearly defined and quantifiable goals. Yet all of us working in higher education were aware that our goal statements in colleges and universities were often nonexistent or at best highly diffuse proclamations that were likely to be idealized, nonquantifiable statements about character-building, preparation for improved citizenship, or fulfilling broadly stated individual or societal needs. We further knew that our goals, insofar as we had any, were often multiple in nature (such as the Gross and Grambsch's study that utilized 47 goal items for major universities), that we had little in the way of priorities for these multiple goals, and that they often changed rapidly depending on social concerns of the day. Thus, there was bound to be a clash between any rational information or management system design that presumed goals could be easily identified or tied to definitions of program or output structures. In fact, we have seen many institutions struggling with this question and, perhaps, the development of information systems has forced us to look at what we are trying to accomplish more seriously. However, it is fair to say that most institutions are still vague enough about goals at any level to make the job of an information system designer concerned with management applications a very difficult one.

Multidimensional Structures and Incongruities. The second area of concern was the relationship between the information system and the institution's formal organizational structure. While some college structures were defined in terms of supportive processes or functions, such as admissions, registration, etc., the productive structures for educational, research and service activities were colleges, departments, centers, institutes, divisions, etc., often overlapping or not clearly differentiated from each other or the support structures. This meant that designing an information system that related to the formal structure of the institution was going to be extremely difficult. Any of you who have struggled with program budgeting within the context of the institution know the difficulties of distinguishing productive and support units and dealing with the problems of direct or indirect costs. This issue has been further exacerbated as we have made attempts to apply some of our rational information and management notions to the research and service areas.

Nature of Decisions. A third formal organizational area of concern was the question of: 1) whether the information system would provide reports for basic day-to-day operating decisions (admissions, registration, budget reports, and other measures of day-to-day activity), 2) whether they would focus on managerial decisions concerning the assessment of different organizational units on the basis of efficiency as reflected in resource allocation decisions or on the basis of effectiveness along some defined goal notions, or 3) whether the design would focus on broad decision issues related to institutional policy and planning questions. Obviously institutions themselves did not make these distinctions very clearly in their decision-making structure. Further, it was easier to develop information systems to support the day-to-day operating level needs than to design them for the managerial or planning functions or decisions. This has been expressed over and over again by individuals trying to develop reporting schemes and studies to support policy decisions. Policy issues are not always output oriented and are often shifting issues or ones with little advance notice. They often require new internal data or data external to the institution for comparative analysis (e.g., the shift from nondiscrimination to affirmative action).

Effectiveness Ambiguities. The final formal organizational area of concern was the ambiguity about what constituted effectiveness in a college or university which in turn made it difficult for information systems designers or management systems analysts to design approaches that would measure effectiveness. Was it merely quantifiable measures of productivity (SCH or degrees)? Was it measures of efficiency ($/SCH)? Was it more qualitative measures of educational goals or student gains? Was it goal achievement of particular programs? Was it program or institutional impact on the individual or the larger societal sector to which a program was directed? Or was it some longer range notion of programmatic or institutional educational effectiveness? And what are all of these measures for research or service? For the most part attention to date has been directed to quantitative productivity and efficiency measures of teaching activity with only scant attention to goal related measures and seldom to qualitative or longer term notions of effectiveness. Perhaps the best outcome of the progress of this "efficiency cult" is that it forces the detractors (who are more often on the academic side) to face seriously the issues of educational quality which they have so often resisted defining or measuring.

Informal Organization Strains:

In addition to the potential conflict between the introduction of information and management systems and its related technologies with the formal organization, there were four informal organizational dimensions where strains were likely to occur: decentralization, participation patterns, information overload, and norms.

Decentralization Dilemma. First was the decentralization dilemma. While it may be easy to design an information and reporting system when the locus of decision-making is clear and consistent, our colleges and universities have been notorious for having diffuse patterns of decision-making. Not only do they differ from institution to institution, but within large institutions different decisions get made at different levels in the institution. Often the same decision within a major university may be made differently within different colleges or even departments. The same decision may even be made at different levels at different points in time. Thus, the dilemma was clear. Attempts to either centralize or, at least, clarify decision-making levels were necessary to design any regular reporting system that would support decision-making. Yet many institutions tend to resist clarifying points of decision-making and/or to stress decentralization that make the design of information systems that would be useful to administrators very difficult.

Participation Dilemma. Closely related to the decentralization issue was the question of the impact of pressures for increased participation in decision-making on the accommodation of management oriented information systems. As participation increased, obviously data sensitivity became more of an issue, and questions of clearance and access would become more difficult. Also enhanced participation, whether by faculty or students, would introduce many less sophisticated people into the decision-making process who would need more explanation or elaboration in the kind of reporting information that was developed. Simultaneously, increased participation often leads to raising new issues or new questions which can increase the demand for management information. At the same time opposing groups may be using information for their own political struggles. This places information system offices under the potential stress of demands for additional information possibly by less sophisticated users as well as finding themselves in the center of political controversies over whose information should be developed first and whether any information collected was legitimate or accurate in the first place.

Information Overload. The third potential informal implication of expanding information and management reporting systems in the college and university setting was the likelihood of creating an information overload for administrators already inundated with paperwork. The problem of designing information systems which recognized the existing, but chaotic information system, and then clarified or simplified it and reduced reporting was at least as important as developing more information in many institutions. That, however, was often a less exciting part of information systems development that was overlooked.

Diverse and Conflicting Norms.. A final area of difficulty in a college or university's informal organization structure for management information systems development was the fact that they would be thrust into the campus situation where there were often diverse or conflicting norms regarding information and its uses. For instance, the press for "openness" and sharing of information on many campuses and the press for "individual confidentiality" raise serious questions about the structure of information collected, the manner in which it is analyzed, and its accessibility. The recent discussion of the mandated student access keeps this conflict alive. Another issue is the stress between the necessity for centralization in order to be accountable which conflicts with many of the faculty and students' presses for participation and decentralization. Thus, norms and attitudes, shaped partially by larger societal or legal concerns about data and its uses and partially by the local campus cultures, was bound to influence the nature and complexity of information system's development and their utilization on different campuses.

External Demands:

In addition to the internal strains with the formal and informal organizational patterns, the introduction of comprehensive management information systems also potentially affected the external demands on an institution. While it was readily accepted that better information was needed for a wide variety of external reporting requirements, the fact that an information system would be developed which might encourage cross-institution comparisons was likely to raise the politically sensitive questions of institutional autonomy versus external control.

Continuing attempts to develop statewide reporting systems and analytic structures by NCHEMS and other state and federal agencies reflect this as a continuing issue. It is a struggle which is often out of the hands of people involved with development of data and reporting systems on individual campuses. Yet it is one which has a day-to-day impact on their own operations.

These concerns of three years ago that the forms of information technology being developed and the mode of introduction would cause substantial strains within colleges and universities are, no doubt, things you have all experienced and, in some institutions, have partially resolved. You have all been involved in the day-to-day struggles to improve the fit between the systems which are either imported or internally developing and the formal and informal dimensions of your own institution. Perhaps the best indicator that the fit is improving, that academic and non-academic administrators and management information systems people are beginning to learn to live together, is the change represented at the 1973 National Forum on New Planning and Management Practices held in Chicago compared to the one held two years earlier. This second one did identify and deal with some of these issues. Indeed, the conference title was "Information Impact--Collision with Tradition." The attendance itself was more evenly balanced between academic administrators and those from the business administrative or data processing side. At least the "technocrats," as Lyman Glenny had referred to them earlier, were no longer the only group present. But this was the struggle of the first epoch, the development of information systems and related technology for better planning and management systems. And although these issues have not completely abated, there are more important things changing in our current higher and postsecondary educational scene that now should concern us.

THE CURRENT SCENE--EMERGING CONDITIONS

Unlike the earlier issues which revolved around the development and the attempted initial introduction of the information systems technology itself, there are on the horizon a new set of dynamics or the emerging conditions which threaten once again to isolate the information system personnel from the non-academic and academic administrators and decision structures of the institution (and perhaps even to remove them from the campus) just at a time when a more effective integration of the information systems personnel into the academic and administrative decision-making life of the campus was occurring. In keeping with the initial allegory, there are seven new conditions that are identified. In discussing each, I will suggest their possible impacts on information systems development, the implications for the relationship of information system staff or designers with faculty or administrators, and some of the concerns that this raises for future information systems directions or staff development activities. The first two are developments which may be strengthening the relationship of information system personnel to the academic and administrative functions on campus, so to the good news.

Improved Technological Innovation and Introductory Strategies:

First, after the initial period in which many of the information systems developments were designed or initiated externally and often implemented under the direction of a few administrators or systems experts, we now seem to recognize some of the difficulties in introducing a comprehensive new information technology into colleges and universities. We have learned, like business, government and the military before us, that it is often necessary to redesign some of these rational, externally designed systems and applications to fit our own campuses. Also many institutions are recognizing the need to develop advisory groups with broad representation to guide information systems developments. It is also more prevelant to find some internal institutional training programs for department chairmen and other lower echelon academic and non-academic administrators in the developments or uses of information systems and its related hardware and software. These refinements of the earlier top down or externally imposed strategies of introduction seem, on a number of campuses, to be improving the relationship of information systems staff with both the faculty and academic administrators as well as the non-academic administrators.

The clear implication for information sytem staff, however, is the need to learn to work with these seemingly slower or more complex groups in setting directions for or gaining acceptance of proposed information systems developments, policies, and the like.

Administrative Staff Changes and Developments:

Another emerging condition that affects the developments of information systems in colleges and universities are changes in the administrative personnel and related information systems staff. In the six or seven years of publicized activity in this area, there has been a substantial period for staffing turnover in both the information and data processing staff and in the academic and non-academic administrative staff of most institutions. Many of our more recently appointed academic administrators have greater awareness of the potential of computer and information systems developments and applications to help them better manage their institutions. At least there is enough of an awareness that they have added assistants to their own staff who have such knowledge if they themselves do not. On the information system side, many of the top positions are beginning to go to persons who have had some years of experience working in colleges and universities. They now come to the task of developing information systems not with just the business oriented approaches and experiences that they brought with them from industry and government but with a greater awareness of some of the dimensions of educational organizations that I have previously discussed. Unfortunately there is still a greater emphasis in many places on esoteric hardware and software development than on client (student and faculty as well as administrative) needs which will be discussed later. These important developments in staff background, experience, and attitudes certainly have strong potentials for improving the working relationship among information systems staff, academic and non-academic administrators, and faculty.

An additional important development in many institutions is the establishment of a planning officer who may work closely both with the academic and administrative officers of the institution and with the data and information systems directors and the institutional research or studies offices. Some of the personnel in these new planning offices have substantial expertise in integrating the academic and administrative problems and issues with the information system and technology. Thus, they constitute an additional bridge to improve the integration of data systems developments into the mainstream of college or university decision making.

The message for those concerned with developing information systems is clear. There is a need on the information system side for personnel who have a good understanding of the educational institutions in which they are working and, similarly on the administrative side, for staff who are aware of the developments in information systems, computer applications, and management techniques. The institutional potential of a planning officer who is an effective bridge of course cannot be overlooked.

Growth of Collective Bargaining:

A third development in higher education that is likely to affect the development of information systems substantially is the continuing growth of collective bargaining. We currently have permissive legislation in approximately 20 states with an extensive pattern of bargaining in community colleges and the public four-year institutions in many of those states. The continued press of inflation is likely to continue or exaggerate that growth as faculty, even in prestigious institutions, become more aware of the needs to protect their own salary gains from those of other, more organized groups on the campus. In declining institutions the need may be even more severe just to protect their positions. In institutions that have gone to collective bargaining one dynamic is clear. The administration puts a great deal of effort, if it has not done so already, into expanding its data base and orienting it very clearly to management and resource allocation issues. At the bargaining table good resource data is necessary in order to do an effective job of negotiating.

The tendency of this kind of data base expansion and analytic orientation would seem to improve the relationship between the information system staff and those in the business and financial administrative or employer roles. It is not immediately apparent that it would do anything other than isolate them to a greater extent from the faculty who will be more and more suspicious of various attempts to collect data. Information systems staff, of course, need to be aware that this is a shift from an emerging and often desired neutral stance to a clearly administrative orientation and to the fact that much of their work is going to be politically sensitive in the area of negotiations between the faculty and administrators. It also means that some of the information systems staff need to be more sophisticated in working with financial and resource issues than ever before.

Social Issues on Campus:

A fourth emerging condition is one that threatens to isolate the information systems staff from both faculty and administrators to some extent. This is the growing concern on campus for social issues, such as affirmative action, that have an extensive legal or governmental enforcement based impetus. One of the clearest directions of affirmative action guidelines and affirmative action programs on various campuses has been the necessity of substantially expanding the personnel records and reporting requirements in an institution. Not only are the records themselves expanded but many of the policies and procedures for recruiting, selecting, evaluating, and promoting academic and administrative staff at all levels tend to become much more routinized or bureaucratized. A large part of this impact falls in the personnel or affirmative action offices which will inevitably work closely with and depend heavily on various data processing units to develop information systems and analyses to support the information needs and demands of this area. While faculty and administrators on many of our campuses may be sensitive to concerns about affirmative action, they are often more threatened or resistant to many of the extensive new information requirements and procedures which they have to go through. This potentially may isolate the staff involved with the development of such machinery on campus from some areas of administration and academic life. Further, the appropriate interests of many of our ethnic minority and women's groups is to assure that such information does get developed and is widely available. Thus, much of the information generated around personnel records is politically very sensitive. Simultaneously, federal guidelines opening student files or the proposed "Buckley Amendment" prohibiting race or sex data further complicate the future direction of this area.

The implications for persons involved in the development of information systems in these areas would suggest not only being aware of the type and extensive nature of the information demands of affirmative action and of related legislation but also to encourage the development of a broadly based advisory group to provide guidance in the development of personnel data information guidelines, reporting and analytic needs, and the various policies regarding its use. There is a need for such an integrating group both to protect the interests of minority groups and to protect the information systems offices from the potential isolating effects of being viewed as the developer of the affirmative action enforcement machinery if not the enforcing agent itself.

Declining Enrollment and Economic Conditions:

Although the discussion of declining enrollments of the past two years is slightly abated this year and we have become accustomed to the seemingly perpetual financial crises, anyone working on a campus today is aware that enrollments are still declining in many institutions and that inflationary pressures are making budgeting allocation decisions more and more difficult. The reduced institutional flexibility or slack resources brought on by financial and enrollment pressures takes away many of the positive arguments for information systems development that are managerial in character. Many of the early arguments for developing improved or comprehensive information systems and better planning and management strategies based on rational analysis was that it would provide us with means for making better decisions for improving our campuses. Yet, faced with enrollment declines or constantly delimited financial situations, information systems and related analytic studies are used more often to rationalize reductions and cuts rather than to justify new programs, new directions, or reallocations. In a sense, if we had developed planning and management approaches and the supporting information systems in the 1950's and early 1960's, we would have had an era in which we could have made positive arguments about the new things that were accomplished as a result of our better informed decision-making processes. In our current situation the uses of data are likely to be viewed negatively, especially by faculty and lower level academic administrators. Thus, the tendency is to drive the information systems staff of the institution further from the academic staff. The non-academic and top level administrators may be attuned to dealing with these difficult issues so relationships with them may be improved.

The implications, however, for staff involved in development of data systems and analysis activities is that they must recognize that they may be viewed as a participant in some very difficult and unpopular institutional decisions. In view of this situation they may need a thicker hide as long as these conditions continue. They should certainly be aware that the promises about the positive values of information systems developments is more likely to be perceived as rhetoric than reality.

New Uses and New Image--MIS or CIS:

Sixth, there are a number of interrelated developments in higher education today that may change our recent notions of information systems as predominantly management information systems to client information systems. We do, of course, have information and computer systems that provide research assistance to faculty, library information to students and faculty, national clearinghouses like ERIC. [CAUSE is even a client information system.] However, the current decline in enrollments combined with the shifting of federal aid from institutionally based funding to student based funding has encouraged a greater market orientation on the part of institutions of higher education. This movement is supported by recent developments in and popularity of new alternative educational forms which emphasize individualized, self-designed programs and self-paced instructional materials for traditional and nontraditional students. Whole institutions are conceived as new academic delivery systems such as Empire State or Minnesota Metropolitan. These developments suggest an extensive potential need for data systems that are oriented towards students who need to have an assessment of where they stand, who need to have an information system that provides them and their advisors with information either on the various educational alternatives and materials they might use in seeking individual educational goals or who need to find other learners with mutual interests and concerns. While we have utilized some information systems to deal with students in terms of admissions, registration, and testing for counseling and advising, those have been essentially institutional based orientations. What we are now talking about is students who enter an institution like Empire State, develop their own educational goals, learning contracts, and educational programs which can be accomplished through a wide variety of educational modes, and who then need an extensive information resource to help them identify the various learning modules or options that are available to them. In educational modes such as this, a computer based information system directed to students might be a very efficient information exchange network. Our increased concern for adult and continuing education and the use of other forms of off-campus learning may further expand this demand.

A related development is the potential applications for computer based learning systems such as the PLATO system at the University of Illinois, which is currently being evaluated. The development involves a central educational hardware and software system that allows faculty from remote terminal locations to author courses which are then available in the system to students or faculty who may elect to use them in the computer based learning network. Their array of subject matter and levels of learning is impressive despite its limitations. In a sense it is an example of an information system designed for the use of both faculty and students as clients in the educational process. But it has management systems applications in the sense that much of the cost and utilization data can be built into the system itself.

In another development the use of computers as a resource for student and faculty and more recently administrative researchers is being widely accepted. Spurred by EDUCOM experience, computer networks such as ARPANET span the globe--connecting computers on various campuses via telestar. More parochially, institutions are now sharing their research computer facilities with simpler networks.

The MERIT system in Michigan connects the University of Michigan, Michigan State University, and Wayne State University computers and further makes those facilities available to smaller institutions on a time-sharing basis. Within institutions, administrators or their data analysts who possess an increasingly sophisticated knowledge of computer technology and information systems may use their data base in less routine and more immediate modes to analyze issues with on-line remote terminals.

Other client based computer information system developments have potential application in governance. For instance, on our own campus Professor Merrill Flood, who has had a substantial background in management sciences, is currently working with our university faculty to improve communication through the use of various kinds of telecomputer and teleconferencing networks which would allow faculty to be consulted through remote terminals on a "store and forward" basis for governance issues on our own campus or to participate more rapidly in Delphi-based goal-setting exercises. He also is exploring ways of using teleconferencing and a telecomputer network to bring together researchers from wide and diverse points for research or other educational issues. While extensive governance-based applications such as Professor Flood's may be a long way into the future, it is obvious that there are many student and faculty, educational and research as well as administratively based needs that can still be served by computer based information systems.

Many of these client based applications seem to involve extensive computer networks and sharing of arrangements which will have substantial implications for those in charge of computer information systems in colleges and universities. Indeed, experts in telecommunication may be our next wave of on-campus technical experts. Many of these client-oriented applications also cut across some of the artificial barriers that have been created between administrative data processing and educational or research applications. Data systems developers are in a position to potentially integrate their efforts not only with the administration but also with faculty and student interests. If they respond, they may create a new image for themselves as people who are responsive to a broader range of users. This new image for staff involved in information systems development suggests that they need to be developing flexible, accessible, and nonroutine information systems rather than the more heirarchical or stable models that sometimes serve management information systems needs. It also suggests the use of staff with a broader range of telecommunications as well as information systems development skills. There is a need for individuals with experience in designing new research and learning systems from a wide variety of fields. It further suggests turning to information processing designers who have worked in health and social welfare agencies, hospitals, government employment areas and other kinds of social institutions which are much more client oriented and who have developed models of client based information and communication systems. On many campuses this kind of a reorientation would be both an extensive and difficult one, requring commitments of many different administrators and faculty and not just the information systems staff itself.

Statewide Computer Plans and Networks:

A final development, and one with which many of you are already familiar, is the recent growing tendency of states to develop statewide computing plans or networks. Currently, according to a study by Charles Mosmann for EDUCOM, 16 states have already developed statewide plans for computer development or use and 15 more are in the process of developing the same. The reasons for it are clear. Legislators and other state level personnel are concerned about the high costs of computer hardware that is often duplicated on a variety of campuses. They are also concerned about delivering computer based management information, educational, and research services to many of the smaller campuses that do not have extensive facilities: the technical expression of "equal access." But one effect of some of these plans is to centralize, in a limited number of locations, the computer hardware that provides administrative, educational, and research services for the state higher educational system. In some states it is apparent that even much of the software application developments would reside in these more central locations, thus reducing the information systems personnel on individual campuses to data collection, data retrieval, and assisting others in using the central system from on-campus terminals. While the exact direction of these developments is unclear, there is currently one study being pursued by two graduate students in our Center (Richard Heydinger and Donald Norris). It has EDUCOM sponsorship and is expanding Charles Mosmann's survey of state level computer activities. They are looking at the nature of the statewide computer plans, their implied organizational governance structure, the patterns of institutional involvement, and arrangements of hardware and software services. The initial implications are that the plans involve various telecommunications media as well as computers: audio-visual modes, information systems, and computer assisted applications for instructional research and administrative activities. Regardless of the directions, it does mean that institutional staff involved with computer based information systems need to be more aware of the development of statewide computer or communication plans and may even need to look closely at new cooperative mechanisms of developing computer based networks that serve several campuses and involve other communications technologies. Further, it is necessary to follow closely what is happening in statewide agencies and legislatures as these plans take shape if long-range developments on individual campuses are to be realistic.

SUMMARY AND CONCLUSION

In summary, this paper has covered a great deal of material. A review of the initial period of information systems development suggested several ways in which the development and introduction of this expanding technology created strains with both formal and informal dimensions of college and university organizational and administrative structures. It was suggested that attempts by campus administrators and information systems developers to deal with these strains were progressing and thereby reducing some of the initial negative perceptions of information systems staff and personnel as technocrats or management oriented centralizers with overly rational or simplistic views of the applications within colleges and universities. However, it appears that there are a new set of conditions that are facing us--conditions that no longer result from the pattern of development and implementation of the technology itself, but rather conditions that are emerging outside of that technology that have potential implications for information systems staff and their relationship to the educational and administrative personnel in colleges and universities and other communications technologies. The problems created by the "first epoch" of information systems design were being resolved and information systems, even their management applications, were becoming legitimate and integrated into colleges and universities' academic and administrative patterns. But now many of the new conditions threaten that improved relationship and require astute leadership both from the administrative and academic leadership in colleges and universities and from the administrative staff of various information systems offices who must develop and expand their own role as well as those of their staff along possibly new dimensions. Those changes include the need for a greater awareness of innovative strategies for developing and introducing the new information technologies; better selection or development of the personnel who can be interstitial, who can integrate the interests of the information systems and computer staff with the educational and administrative staff; greater sensitivity to the political nature of information in both collective bargaining and social issues that are impressed upon the campus; the need for toughness in dealing with institutions where many of the applications of information are perceived by many on the campus as negative; the potential of a whole new image of information system people oriented towards client information systems as well as management information systems; and finally the need to be aware of developing statewide plans for computer and information or communication system services and their related new network and sharing arrangements. All of these dimensions will vary from state to state and campus to campus, yet they seem to be ones that require consideration if the designers and purveyors of information systems are to continue to effectively integrate their efforts with academic and administrative staff and with the real decision-making patterns in institutions of higher education. As with any allegory, I leave you neither with optimism nor pessimism but hopefully with insights and challenges that you can translate into your own setting and future development work.

EMERGING CONDITIONS AFFECTING INFORMATION SYSTEMS DEVELOPMENT AND INTEGRATION IN HIGHER EDUCATION

Emerging Conditions	Implications for I.S. Development	Implications for I.S. Integration	Some I.S. Staff Concerns
1. Increased emphasis on introduction strategies.	a) Redesign of external systems. b) Less top down emphasis. c) More internal I.S. training.	Improved integration with academic and non-academic administration.	Awareness of institutional uniqueness. Greater use of participation and educational strategies.
2. Administrative staff changes and developments.	Greater mutual understanding by I.S. and administrative staff.	Improved integration with academic and non-academic administrators.	New personnel selection criteria--educational and/or interstitial experience.
3. Growth of collective bargaining.	a) Expand and routinize data base. b) Stronger resource allocation orientation.	Integration with non-academic administrators; isolation from academic admin. and faculty.	Sensitivity to political and resource role. Reduction of image of neutrality. Increased load.
4. Social issues--legally mandated.	a) Expand and routinize personnel data base and procedures. b) Image of "enforcer."	Isolation from academic and non-acad. administrators unless well handled.	Awareness of political and personal sensitivity of data. Expanded data needs. Need for advisory structure.
5. Declining enrollment and economic conditions.	Negative uses of data and analysis.	Isolation from faculty and lower admin. staff. Integrate with top admin. staff.	Reduction of rhetoric about positive uses of data. Personal toughness.
6. New image: CIS vs. MIS.	Emphasis on students' and faculty educational and research needs. New governance applications.	Integration with faculty and students. Possibly with administrators.	New I.S. design orientation and models. Broader I.S. staff expertise. Great use of networks and sharing arrangements.
7. Statewide computer plans.	a) New supra-institutional groups. b) Centralization of hardware and software. c) Networks.	Possible isolation or removal from campus relationships.	Awareness of state plans. Increased use of sharing and networks.

THE CASE FOR TOP LEVEL INVOLVEMENT IN MANAGEMENT INFORMATION SYSTEMS

Dr. Ronald W. Brady
Vice President
Planning and Resource Allocation
University of Illinois

THE CASE FOR TOP LEVEL INVOLVEMENT IN MANAGEMENT INFORMATION SYSTEMS

Successful companies have long recognized the need for top level involvement in management information systems development. The top administrators, the presidents, provosts, and vice-presidents of only a few colleges and universities have devoted attention to computing affairs; and then usually only when a crisis erupts. Dr. Brady discussed the need for and the methods of involving top administrators in management information systems development.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

REQUEST FOR PROPOSALS: FINANCIAL SYSTEMS AND HARDWARE

John F. Chaney
Director
Management Systems
University of Colorado

James S. Martin
Analyst
Management Systems
University of Colorado

Request for Proposal (RFP) to Define Financial Systems Requirements of the University of Colorado

John F. Chaney, University of Colorado
James S. Martin, University of Colorado

BACKGROUND AND DEVELOPMENT

In February of 1974 the Regents of the University of Colorado decided to engage a professional firm to provide consulting services to the University. As defined by the Request for Proposal (RFP), the consulting firm was to undertake a project which would conceptually define financial system requirements for the University's four campuses, including a comprehensive Medical Center campus with a teaching hospital. Several CPA firms as well as software houses were asked to submit proposals in response to the RFP. The Request for Proposal asked that vendors provide consulting services to accomplish the following:

1. Document existing financial systems in operation at the university.
2. Define the new systems required
3. Examine existing accounting systems which would meet the needs of the University of Colorado.
4. Work out an approach and a timetable to implement the financial systems recommended.

There were three major reasons for the development of the RFP. First, there were external pressures, particularly from the outside auditors of the University. The auditors were concerned about significant deficiencies present in the accounting procedures, practices and reporting of the University and with problems at the Medical Center in regard to patient billing and accounts receivable. Secondly, the existing operational data systems were totally inadequate and non-integrated, particularly in their ability to compare accurately budget and expenditure information. Finally, there was a requirement for better program and financial management information necessitated by changing needs of the institution which the existing financial data base was unable to provide.

A management team consisting of senior executives of the University was created to oversee the development of the financial systems project. This team was also charged with selecting the most qualified vendor.

Four basic assumptions were used in the development of the RFP.

1. A simplified version of the RFP was prepared to briefly explain and spell out the University of Colorado's expectations without covering every detail. The University provided sufficient latitude for response to provide the details of what was to be done.
2. Selection of the vendor was to be based upon the expertise of the individuals involved and their experience in designing accounting systems for institutions of higher education.
3. The RFP specified an upper cost limit.
4. The selection was to be made by the management team using rules for selection which were included in the RFP.

CRITERIA FOR EVALUATION OF PROPOSALS

All proposals submitted were evaluated by the management team of the University of Colorado. Criteria used for selection included the following general categories:

1. Experience of the assigned individuals with similar work at institutions of higher education including institutions which were major public

research multi-campus universities which had associated medical schools and teaching hospitals.

2. Previous client engagements, including appropriate references.
3. Overall experience and background of the assigned individuals in accounting and the design and implementation of accounting methods and data processing systems.

The complete list of criteria for selection of the vendor is contained in Appendix 1.

DESCRIPTION OF THE REQUEST FOR PROPOSAL

The request for proposal consisted of four major sections: general information, proposed system requirements, information required from vendors as the result of study, and format guidelines to be used in preparing the RFP.

GENERAL INFORMATION

The General Information Section contained primarily procedural information including the purpose and scope of the RFP, schedule of activities, method of submission, acceptance time for the University of Colorado, a detailed description of the criteria for the selection of the vendor, and provisions for service beyond the scope of the specifications. In addition, the RFP noted that the University intended to provide the equivalent of two full-time staff to the project in order to provide long-term continuity and benefit to the University of Colorado.

PROPOSED SYSTEM REQUIREMENTS

Respondents to the RFP were asked to define and document existing financial systems to develop an understanding of the existing financial systems. This was necessary to provide background to determine what elements of the old system must or should be included in the new system. The suggested methods and approach to defining and documenting existing systems as specified in the RFP included:

A. Review and analyze the existing fund structures and chart of accounts. Compare and integrate these processes as found on all four campuses.
B. Review and document the present budgeting system. Analyze existing data areas to determine relationships between budgeting and accounting systems.
C. Review and document the accounting principles currently being utilized on all four campuses.
D. Review and document the methods of accummulating, recording, reporting and retaining accounting and financial related information. This includes the general ledger and all books with original entry, i.e., inventory disbursements, receipts, payroll, encumbrances, accounts payable, etc. Compare the similarities and differences among the campuses.
E. Review and document responsibility for both maintaining and using the budgeting, accounting and financial information at all levels of responsibility.
F. Where possible, use the documentation already gathered by the University of Colorado.

Vendors were also asked to define new system requirements in order to determine the management planning and control needs as well as the general custodial and third party reporting demands of a financial system. The methods and approach specified in the RFP to accomplish this task included the following:

A. Review and document internal management's information requirements for planning, budget preparation and justification and controlling University functions. Special program characteristics of the Medical Center were to be given particular attention.

B. Discussions with all levels of management from the Board of Regents to various department heads and with those people responsible for the custodial or fiduciary aspects of the systems.

C. Examine and document the information requirements of such third party areas (not related to the State of Colorado) as grants and contracts. Determine and document the changes required by the new American Institute of Certified Public Accountants (AICPA) audit guide, the American Hospital Association (AHA) and the National Association of College and University Business Officers (NACUBO), e.g., encumbrances, cash basis or accrual, investment accounting and the use of accounting principles.

D. Review and document the requirements of the various State of Colorado Departments responsible for some aspect of the University of Colorado. These would include the State Controller's Office, the Executive Budget Office, Colorado Commission on Higher Education, State Auditor and others as determined by the study.

E. Determine and document the specific requirements for budgeting, financial, program and work load information for the University's operations, in particular the Medical Center.

Vendors were also asked to examine the existing State of Colorado accounting system, other university accounting systems and accounting systems available from systems and software developers. The objectives of this task were to review and understand the concepts, systems and resulting information related to the systems examined; to review the advantages and disadvantages to the University of Colorado of the systems examined and to compare the elements of the accounting systems examined with the needs defined in the definition of new systems requirements and to note any anticipated problems or deficiencies if any. Vendors were also asked to prepare recommendations for adopting all or a portion of the system believed to be most suitable with the needs defined in the preparation of new system requirements.

Finally, potential vendors were asked to recommend a detailed approach and timetable to implement the system recommended.

INFORMATION REQUIRED FROM VENDORS AS A RESULT OF THE STUDY

The RFP presented information requirements which constituted an outline of what the University of Colorado believed to be the major processes for completion of the study. It was also indicated that additional information should be furnished as the vendor believed necessary in order to cover the study properly.

The results of the definition and documentation of the existing financial system were expected to provide the following:

A. Flowcharts of existing systems.
B. Definitions of files.
C. Documentation of transaction volumes.
D. Description of input and output documents.
E. Summary of functions performed by personnel.
F. Description of the accounting principles, fund structures and chart of accounts.
G. Organization charts including identification of levels of management preparing and using financial information.
H. Summary of existing systems constraints such as report timing and special third-party requirements.

The results of the definition of systems requirements were expected to provide the following:

A. Definition of accounting principles to be used.
B. Conceptual designs of operational and management output reports.
C. Chart of accounts and a fund structure.
D. General Ledger systems design.

E. Conceptual flow chart and schematic diagram of an integrated financial system including definition and purpose of the various systems modules.

F. General definition of program planning concepts.

The examination of existing accounting systems available from other institutions, agencies or software developers was expected to provide recommendations for adopting all or a portion of the system including the pros and cons of each.

It was also expected that a total plan for implementation would be developed to support the recommendation along with a timetable for implementing the proposed system.

FORMAT GUIDELINES

Format guidelines requested that the proposal be submitted in eight major sections.

1. Management summary.
2. Scope of engagement and project approach.
3. Project Schedule.
4. Consultant firm background and other relevant experiences.
5. Proposed project organization and staffing.
6. University support requested.
7. Certification as an Affirmative Action Employer.
8. Cost data for proposal which was to be bound separately from proceeding sections and sealed.

ORGANIZATION OF PROJECT AND DELIVERABLE PRODUCTS

Figure 1 describes the project organization that was used to complete the requirements study after a vendor was selected. The combination of consulting staff with university staff was found to be very valuable in providing the continuity needed to implement the recommended conceptual design. The deliverable products that arose out of the study are listed in Appendix 2.

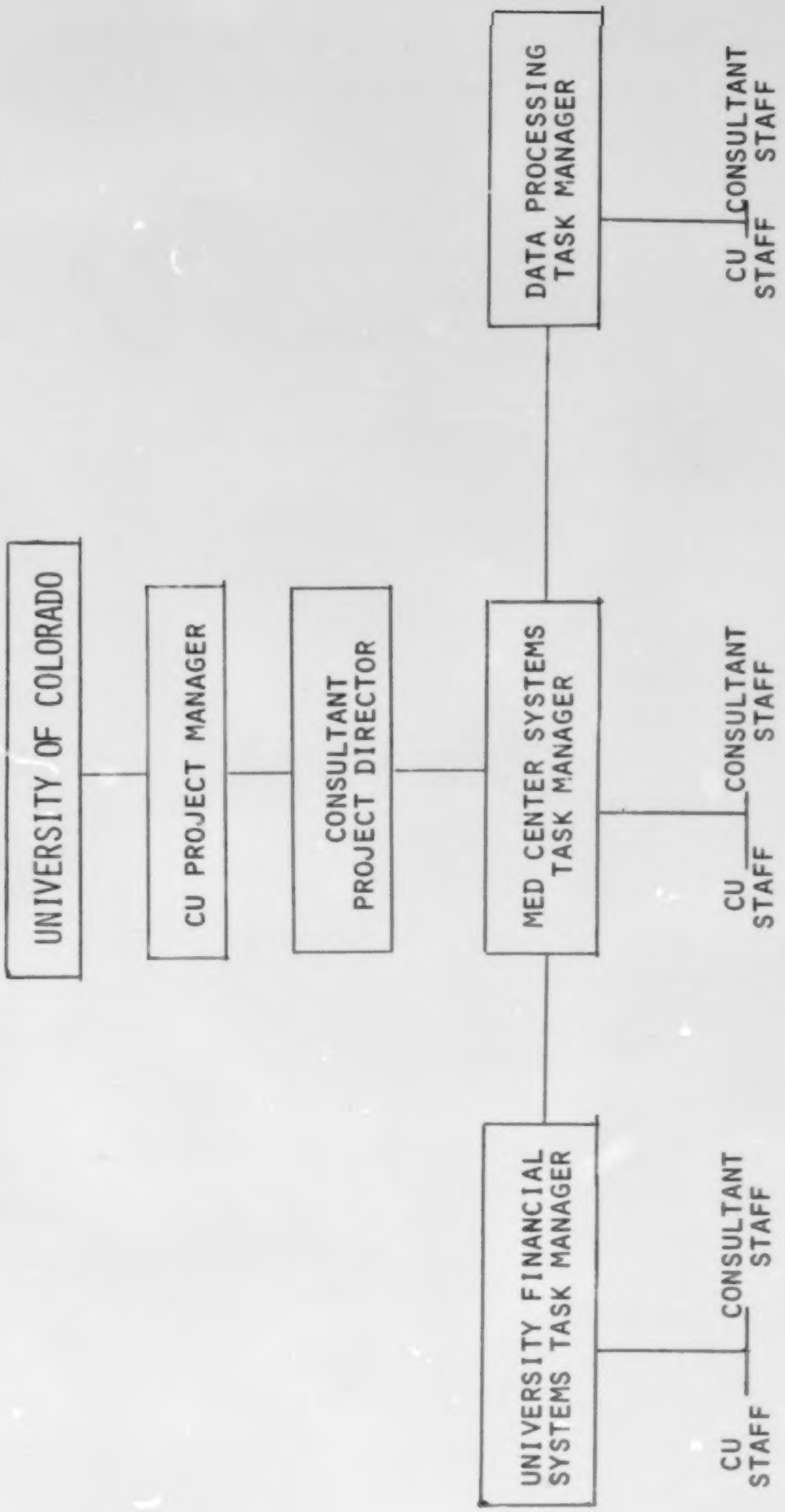

UNIVERSITY OF COLORADO
CU PROJECT MANAGER
CONSULTANT PROJECT DIRECTOR
MED CENTER SYSTEMS TASK MANAGER
UNIVERSITY FINANCIAL SYSTEMS TASK MANAGER
DATA PROCESSING TASK MANAGER
CU STAFF
CONSULTANT STAFF
CU STAFF
CONSULTANT STAFF
CU STAFF
CONSULTANT STAFF

FIGURE 1

CONCLUSION

After completion of the evaluation process, a contract was awarded. The final report was delivered in December of 1974. The report contained the financial systems requirements, conceptual design and the recommended implementation plan for a new University-wide financial reporting system. This report is providing the framework within which the University of Colorado will implement a new system over the next several years.

APPENDIX 1

CRITERIA FOR SELECTION OF VENDOR

A) The experience of assigned individual(s) in the same type work for an institution of higher education.

B) The experience of the assigned individual(s) in similar work for an institution comparable to the University of Colorado according to the following:

1. Public institution
2. Graduate instituion
3. Medical Center
4. Multi-campus institution
5. Equal or larger operating budget
6. Comparable number of colleges

C) A list of client public institutions of higher education including:

1. A description of the work performed for each
2. Address of references
3. Vendor personnel assigned to each

D) The overall accounting experience of assigned individual(s) other than in higher education.

E) The experience of the assigned individual(s) in the design an implementation of accounting methods to data processing systems.

F) Client references.

G) Vitae of the assigned individual(s)

H) The cost which the vendor proposes for the completion of the study. The proposal will set forth the separate cost of each of the sections and the University of Colorado reserves the right to accept, reject or modify any or all of these sections.

I) Vendor's proposed study completion date, list and timetable of checkpoints to evaluate progress.

J) Documentation, letters of substantiation or copies of similar work completed for comparable institutions of higher education.

K) The vendor's requirements for support to be provided by the University of Colorado including:

1. Supplies and cost thereof
2. Space facilities
3. Special equipment
4. Clerical staff and estimated hours
5. Secretarial staff and estimated hours

L) Vendor certification in writing to the University of Colorado as an Affirmative Action Employer.

M) Vendor assumption of prime contractor responsibilities in the event of sub-contracts for portions of the work to be performed.

1. The University of Colorado will consider the selected vendor to be the sole point of contact with regard to contractual matters with full responsibility therefor.

N) Proposal satisfies mandatory requirements.

APPENDIX 2

DELIVERABLE PRODUCTS

A) Semi-monthly progress reports including:

1. Accomplishments during period
2. Current status in relation to schedule
3. Problems or delays experienced and approaches to resolution
4. Next reporting period planned accomplishments
5. Trends and other items of interest

B) Monthly project management reviews

C) Other project reports

1. Final project workplan
2. Short term accounts receivable billings improvements report
3. Administrative Data Processing master plan
4. Present systems documentation
5. University requirements analysis and conceptual design
6. Medical Center requirements analysis
7. Medical Center conceptual design
8. Medical Center implementation schedule
9. Medical Center cost/benefit analysis
10. Software evaluation
11. Final report

SELECTING A NEW COMPUTER

Don Mittleman
Director
Computing Center
Oberlin College

On Selecting a New Computer

The selection of a computer is a process dependent upon many factors. However the process proceeds, ultimately the monetary factor will govern. It is important to recognize the total costs involved; not only the obvious initial costs of the acquisition of the new hardware and the immediately attendant software and the continuing costs associated with the maintenance of hardware, software, and updating of the facility, but also the personnel costs that must be budgeted to continue the infusion of viable applications. It is from this trunk that the branches grow.*[1]

I. Establishing the need:

The environment in which the computer is to function should be carefully defined. Even within a seemingly narrowly defined field such as education, there are variations which need be specified. It becomes necessary therefore for each organization to study its own identity and to prepare a definition of its needs. The more thoroughly and clearly this is done, the easier it will be for the prospective vendors to configure suitable sets of computer components to meet these needs, but more importantly, the easier it will be to evaluate the proposals submitted. By way of example, the following study is illustrative. At a medium size university (6,000 undergraduates, 1,000 graduate students) the search for identity took the form of a marketing survey. A questionnaire was distributed to all faculty and staff soliciting information on their current usage, which was batch oriented, and on their expected usage for the next ten years. While ten years was

Table I
Peripheral Equipment Needs

Equipment / Department	Art		English		Philosophy		Sociology		Architecture		Electrical Engineering		Mechanical Engineering		Accounting		Marketing		Biology		Geology		Mathematics		Chemistry		Physics		Computer Science		Total	
	R	E	R	E	R	E	R	E	R	E	R	E	R	E	R	E	R	E	R	E	R	E	R	E	R	E	R	E	R	E	R	E
CRT *																																
Vector	0	0	0	2	0	0	0	5	0	0	0	0	0	0	0	0	0	0	3	7	0	0	0	0	1	0	0	1	2	0	6	15
Light-Pen	0	0	0	0	0	0	0	0	0	0	0	2	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	3
Hard-Copy	0	10	0	0	2	0	0	4	0	2	4	3	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	21
Rotating Perspective	0	5	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	10	2	8	4	7	11	41
Color	1	4	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	2	6
Remote Conversational Terminal With	0	6	0	5	0	0	0	4	0	8	2	8	3	4	1	5	1	8	0	0	0	0	0	0	0	0	0	0	0	0	7	48
Paper Tape (Cassette) I/O	0	3	3	3	1	4	0	3	0	14	2	6	2	7	1	5	1	2	4	10	0	7	3	2	4	12	2	9	6	16	29	103
Remote Card I	0	5	3	3	2	3	0	5	0	12	7	10	6	5	0	0	0	0	0	0	0	0	0	0	2	5	1	1	1	0	22	49
Remote Printer	0	5	3	3	1	4	0	6	0	14	7	10	6	6	0	0	0	0	0	0	0	0	0	0	1	1	0	0	3	2	21	51
Optical Character Reader	0	9	3	4	1	2	0	7	0	2	0	1	0	1	0	0	0	0	2	0	0	0	0	0	1	0	1	0	0	0	8	26
Card or Tape Output	0	0	1	0	0	0	0	2	0	1	1	2	2	3	0	0	0	0	0	0	0	0	0	0	2	5	2	5	0	0	8	18
Scanning Digitizer	1	2	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	1	2	0	1	0	0	0	0	0	0	0	0	2	13
Real Time On-Line Monitored Experiments	0	0	0	0	0	0	0	0	0	2	0	4	0	0	0	0	0	0	0	0	0	0	0	0	3	4	3	1	0	1	6	15
Two-Dimensional Plotting;	0	0	0	0	0	0	0	3	2	2	1	3	1	1	0	0	0	0	1	2	0	3	0	0	3	0	3	0	2	0	13	14
Movie Making	0	1	0	0	0	0	0	1	1	0	0	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	5	3
																															*	
Alpha-Numeric CRT	6	0	0	2	0	0	0	0	0	0	1	4	1	3	1	2	2	1	0	0	1	0	0	0	0	0	0	0	3	0	15	12

probably too long a period for which to plan, the time from the initiation of the survey to the installation of the new equipment was estimated to be of the order of two years as a minimum. When coupled with the fact that the presence of the newer equipment will undoubtedly generate problems and create demands beyond those originally planned, and this in turn will consume resources at a rate higher than planned, a useful life of about six years may be expected. With the current rate of invention and innovation within computing, and the attendant obsolescence, six years is not an unreasonable period for which to plan.

The response of the faculty and staff was, as expected, in proportion to their current involvement. In those areas and disciplines currently heavily involved, 100% response was expected and achieved; in those areas and disciplines where computer involvement was sporadic or not realized, the response dropped to about 15%.

The survey should consist of more than a mailing of a questionnaire to individuals identifiable as potential users. A personal follow-up to review the information submitted or the reasons for failure to return the questionnaire is indicated if the full involvement and support of the users are to be obtained. In some instances potential applications had been dismissed either because the long range significance of the commitment was not appreciated or because the cost had been improperly evaluated. In particular, the potential for sharing of resources across disciplines had to be developed.

From the user's articulated needs, hardware and software parameters can be approximated. Table I illustrates one method for organizing the data representing the needs for peripheral equipment, similar tables should be prepared for storage requirements as well as processing needs. Research areas are designated under R and classroom courses under E; the numbers represent the number of courses or research areas requiring the devices, not the number of devices required.

The table indicates the extent to which the computer has pervaded both the education and research programs; in the fully developed survey it was extended to include administrative needs as well.

The possibilities for sharing devices need still be determined. However, the local geographic and political climate would undoubtedly be of such importance as to be the overriding considerations. Also, to err on the conservative side would not be disasterous; additional peripheral devices can be acquired at incremental costs.

The software requirements are summarized in Table II followed by a sample offering an intermediate degree of detail.

E. Courses Requiring

This heading includes both those courses presently requiring computer solutions for problem solving and those which have a high expectation of computer utilization with the acquisition of the specialized peripheral devices and the possibility for interactive, time-sharing.

R. Only present research needs are included in this category because shifting research interests make forecasting too unreliable.

1. Statistical Analysis

This heading encompasses normal, binomial, Student's t, F and Poisson distributions, measures of central tendency

TABLE II

Software Requirements

	Areas in which software is needed	E. Courses Requiring	R. Research Areas Requiring
1.	Statistical Analysis	115	102
2.	Solution of Integral and Differential Equations	49	34
3.	Information Storage and Retrieval	48	24
4.	Linear, Non Linear, Statis and Dynamic Programming	26	5
5.	Matrix Manipulation	41	6
6.	Mathematical Modeling (Simulation)	83	39
7.	Contour and Graphical Plotting	13	12
8.	Simplified Language	123	25
9.	Structural Analysis	10	4
10.	Financial Analysis	13	---
11.	PERT and CPM Analysis	5	---
12.	Numerical Taxonomy - Numerical Classification Procedures	6	1
13.	Theoretical Curve Fitting	2	9
14.	Linguistic Stylistic Analysis	4	1
15.	Fourier Series Analysis	8	---
16.	Numerical Iteration Techniques	4	1
17.	Solution of Boundary Value Problems	4	3
18.	General Program for Quantum Chemistry	6	2
19.	Spectroscopy Program	4	10
20.	Miscellaneous	29	21

of dispersion, elements of probability and probability sampling, estimation and hypothesis testing, chi-square analysis, analysis of variance, regression and correlation, time series analysis, index number analysis, and multi-variate analysis. Also included are sets, functions, decision making under uncertainty, game theory, queuing theory and Monte Carlo techniques, Markov analysis, and reduction and processing of laboratory and research data.

2. Solution of Integral and Differential Equations

Included under this heading is the numerical integration of higher order differential equations and the finite difference and finite element solution of differential equations. Particular computer usage in the areas of research includes the numerical integration of the matrix equations of motion, a three dimensional energy density function, and the fifth and seventh order non-linear ordinary differential equations in semi-infinite domains.

Computing center records of past utilization of the computer by individuals, separately reported by course load and research interest, show the consumption of the current facility. These figures may be used to form a base for each user upon which to estimate his future needs. While such estimates may not be precise, they do offer the best information available at the time. Table III illustrates a method for collecting and exhibiting these present and future research needs; Table IV extends the search for projected needs to the classroom. The administrative needs would be exhibited in another table.

Table III

Computing Resources (hours:minutes of CPU time)

Department: Physics

Chairperson: A. B. Carter

Researcher	Actual						Projected					
	1972-73			1973-74			1974-75			1983-84		
	Fall	Spring	Summer	Fall	Spring	Summer	Fall	Spring	Summer	Fall	Spring	Summer
T. H. Remm	1:28	3:15	3:45	2:59	:32	2:25	2:15	1:45	2:30	10:00	10:00	10:00
K. L. Gelowe	7:37	7:04	7:09	18:51	16:22	17:22	20:00	20:00	20:00	30:00	30:00	30:00
S. F. Bleer	2:14	:43	2:19	2:12	1:57	2:39	1:45	2:00	1:45	10:00	10:00	5:00
W. L. Grasdon	1:10	:00	:01	1:50	:38	1:24	1:30	:30	1:00	6:00	8:00	5:00
H. J. Dolivier	:32	2:25	:54	2:01	1:27	1:48	1:00	3:00	2:00	6:00	6:00	6:00
Total	13:01	13:27	14:08	27:53	20:56	25:38	8:30	9:15	27:15	62:00	64:00	56:00

Table IV

Course Number												
11	0:07	-	-	0:12	-	-	0:10	-	-	0:50	-	-
67	1:15	7:21	2:41	12:35	-	0:26	2:30	5:00	2:30	5:00	10:00	5:00
123	-	4:15	-	-	5:47	-	-	5:00	-	-	7:00	-
127	-	-	6:32	9:50	-	-	-	-	5:30	-	-	10:00
129	4:29	-	-	2:46	-	3:46	5:00	-	2:00	-	10:00	-
216	:26	:16	1:53	2:07	-	2:05	3:00	2:00	-	5:00	5:00	-
247	2:47	1:44	-	5:01	4:12	-	5:00	1:00	-	8:00	2:00	-
298	2:49	2:01	-	2:33	2:47	-	5:00	1:00	-	8:00	2:00	-
302	3:05	3:22	-	1:55	3:49	-	5:00	1:00	-	8:00	2:00	-
317	4:35	2:45	5:52	5:12	2:33	4:47	8:00	6:00	8:00	20:00	10:00	20:00
398	1:45	2:16	-	-	2:49	1:01	2:00	3:00	1:00	4:00	5:00	2:00
501	2:38	2:01	4:29	2:57	2:45	12:07	3:00	3:00	6:00	8:00	8:00	30:00
Total	23:56	26:01	21:27	46:08	24:42	24:12	38:40	27:00	25:00	66:50	61:50	67:00

II. The Request for Proposal (RFP)

So soon as it becomes apparent that there is a reasonable expectation of acquiring a new computer, it is prudent to notify computer vendors of this expectation. From this time forward, the informal dissemination of the needs to the vendors and their active participation in helping to organize the requirements has the two-fold advantage of getting independent and competing reviews of substantive data being collected as well as giving each of them (the vendors) as much time as possible to organize the proposal that will be offered. The more deeply enmeshed a vendor becomes in this preliminary work, the greater his initial financial commitment, the more anxious he will be to protect this investment. Also, realistically, it may not be possible to write an RFP which is totally unambiguous; giving the vendors access to the persons having input to the RFP tends to improve their understanding.

The formal RFP represents the distillation of the thoughts of many persons. As such, to insure that no need has been overlooked, it should be made available for comment to all members of the organization who may wish to examine it. As indicated, not only will this insure accuracy, but, since in every organization there are competing needs for resources, spokesmen for these competing needs will have had the opportunity to evaluate their own expenditure requirements vis-à-vis this proposed acquisition. While unanimity of opinion may be desirable, it is not necessary; adequate organizational commitment is.

In addition to submitting the formal RFP to the various vendors, it is desirable to follow this with a formal, technical review session to which all vendors are invited. The purposes of this technical review session are many. A senior officer of the organization can welcome them, stress the significance of the new facility to the organization and indicate the commitment towards its acquisition. This will reinforce in the vendors minds that the probability for acquisition is indeed high.

The presence of staff members to both respond to questions raised by the vendors, as well as to insure that proper stress is placed on specific sections of the RFP, enhances the image of commitment to this project.

Assuming that no covert commitment had been made, either consciously or otherwise, the openness of this forum tends to insure a best effort on the part of each vendor. While some reticense is apparent, at least as may be sensed from the questions asked and inferred from those unasked, the effect is salutary.

(A) Introduction to the RFP

The material to be presented in the RFP should, as a minimum, include the current usage and projected needs as ascertained, details on the current hardware configuration, and on the available software. The philosophy of the current mode of operation and the change that is expected with the new equipment should be indicated.

(B) Expected Mode and Level of Operation (Hardware needs)

Following these introductory remarks, details of the expected mode and level of operation, as complete as possible, should be presented. For example, if the computer is to be used interactively, in a time-sharing mode, with remote terminals, the minimum number of ports to be available on installation should be specified. Expansion capabilities, and the time frame, should be indicated. The distribution between hardwired and dial-up terminals should be given as well as the data transmission rates that are expected. Questions concerning the capability of the communications components of the equipment to recognize terminal speed and character set used and perform automatic translation are typical of those to be asked. If there is no compunction toward acquiring some of these components, i.e. the terminals, cabling, etc., from other sources, this should also be indicated.

For central site job entry, the desired operation characteristics of the input/output devices, i.e. card and tape readers, punches, printers should be specified. As for all items for which information is requested, by indicating the operational need, permit the vendor to suggest alternatives which may improve the price/performance ratio.

If remote job entry is either an immediate need or has reasonable future expectation, indicate this. Ask for equipment description and prices for future expansion in this area, including for example the possibilities of multiplexing.

The possibility that much of the future requirements may be met by attaching the acquired computer to either a regional or national network should not be overlooked. It is not unreasonable to request that any additional equipment necessary to do so be indicated and its cost given.

Planning for enhancement of the facility during the time frame under consideration cannot be overlooked. Choices of equipment for initial installation may significantly affect the price of growth. Require adequate information from the vendor about the availability of devices and their prices to support your expected growth pattern. While a computer installation may not have the expected useful life of other capital equipment, its expected useful life may be substantially increased if initial planning allows for expansion.

(C) Software Support

After covering the details of needs as may be met by hardware, another area in which information is needed is software support: (a) operating systems, (b) programming languages, and (c) applications packages. Also, there will probably be one or more specialized areas which will need software support. For example, the more specialized requirements associated with a data based management system may assume significant importance. Adherence to the specifications and recommendations of industry-wide study groups may be weighed in evaluating the proposal. Again, details on the expected use should be given; it will behoove the vendor to be specific in his reply.

(D) Conversion

Of the programs either operational or under development on the current equipment, some or all will have to be made operational on the new equipment. This conversion effort is generally never trivial and can be quite horrendous. Determine from the vendor the support he will offer to accomplish this conversion and weigh this response carefully with other alternatives such as doing the work internally with current staff, hiring additional staff for this purpose, or contracting with a service bureau. The current usage data that had been accumulated will serve as the basis for the reply. Acceptable minima must be specified.

It is at this time that an enhancement to the current operation may be obtained at moderate cost. Programs in related areas may have accumulated over the past years and too little attention had been given to subsume these separate programs into an integrated one which could have such desirable characteristics as reduced running time, be easier for operations, be less subject to external errors, etc. The conversion of existing programs one-to-one may not be significantly less than converting a group into an integrated system. The comparative costs should be determined.

Emulation of the present computer may not be desirable but possibly can be tolerated. The justification for such acceptance would undoubtedly vary from one organization to the next, but should be articulated.

(E) Special Projects

Attention must be called to those special projects currently operational which had required special hardware or software for their development. For example, analog-to-digital and digital-to-analog conversions at very high rates in real time may pose seemingly unrealistic requirements for some vendors. The vendor must be required to indicate how these special projects will be supported on the system he intends to propose.

(F) Maintenance

(i) Software:

The request for proposal should inquire into the availability of software maintenance in the areas of operating systems, languages and applications programming. The specific terms for the maintenance, including such factors as cost, times of availability, location of dispatch points, as well as other factors deemed desirable for proper support should be stated.

(ii) Hardware:

The proposal should specify the costs for various levels of maintenance, including plans for hours covered, provisions for calls out of covered hours, and provision for on-site maintenance.

(iii) Internal Maintenance:

The possibility exists that some installations may wish to consider supporting either software, hardware, or both internally. To do this, personnel will need to be trained; additional needs could include detailed documentation for software, wiring diagrams, special test programs and equipment, spare parts, price lists for components, sources of supply, etc. Advice on training programs and support, as well as associated costs, should be included.

(iv) Third Party Support:

While it may not be appropriate for inclusion in the RFP, it is not unreasonable to ask of a vendor if any other installations using his equipment is also maintaining it or using third party maintenance. Comparative costs should be developed.

(G) General Considerations

Request information on those specifics of the installation which could have a material affect, i.e. special air conditioning and humidity requirements, water lines, power and voltage regulation requirements, special circuit breakers, etc. By offering detailed information on the proposed site for the installation of the equipment, misunderstandings resulting in last minute delays and overruns can be avoided.

(H) Expected Performance

The history of the past utilization of the computing facility has been available. By coupling the statements concerning the expected mode and level of operation with the expected performance characteristics, a picture should emerge indicating the overall expected performance. Perhaps an illustration would suffice. A current computer facility consists of two machines, one dedicated to batch processing, the other supporting interactive time-sharing via remote terminals. The machine dedicated to batch processing runs under two distinct operating systems, one suitable for processing scientific and engineering type problems, the other suitable for processing business oriented type problems. The overall requirements for the new machine are to process both business and scientific/engineering type problems in batch mode when appropriate while concurrently supporting the interactive, time-sharing terminals. Whereas presently programs submitted for batch processing are run consecutively according to an algorithm prescribing priority, it is expected that (for definitiveness for this illustration) three batch streams would be available in the new computer, again with an algorithm assigning programs to each of the streams and allocating priorities for sharing such resources as the central processor(s).

If a mix of problems running on the present equipment was to be presented, such factors as throughput time minus I/O time could be measured and used as one of several criteria to specify the efficiency of the current facility. It could be required that the proposed computer process the same job mix similarly (single stream) and that the resulting timings be no more than some fraction (i.e. 1/2) of the present ones. Since, however, single stream operation is not the desired mode, this same mix of jobs could be run in the three stream mode and at the same time the time-sharing terminals would be active. Data on the resulting processing time for each job in the mix and for the entire mix could be recorded; the degradation, if any, in the response times at the terminals could be measured. From these data, it would be possible to measure the expected performance of the new system; alternatively, expected performance criteria could be thusly specified.

Whether the procedure outlined above or another is used, it is important that the vendor clearly understands how the performance of his installed equipment will be measured. He should be in a position to guarantee this performance. The details of the test, carefully thought through, carefully documented, and carefully presented should be at the apex of all criteria.

(I) Growth

Times of rapidly fluctuating monetary conditions, coupled with uncertainties in the expected rate of growth in the utilization of the system, make long range planning difficult. In order to minimize these uncertainties, commitments from the vendor fixing such items as long-term rental rates, interest and maintenance rates are desirable. Provision for delivery of equipment several years hence, at current prices, protects growth plans. The additional stipulation allowing cancellation prior to shipment offers protection against obsolescence as well as an over-optimistic estimate of growth.

At this point the RFP is reasonably complete. The present situation has been detailed; the future expectations have been expressed; channels for communication have been established to eliminate ambiguities. Two subsequent events are anticipated: (1) receiving the proposals and evaluating these, and (2) entering into the contract negotiations. At this point a digression is appropriate.

At the time of contract negotiations the following posture may be expected on the part of the vendor. First, there will be a general disclaimer of the information presented in the proposal. This generally takes the form of a statement which indicates that the vendor will honor only the commitments specified in the contract and that the language of the contract supersedes all other statements, either written or oral, made by his representative. Thus, those sections of the proposal which are critical must be incorporated into and made a part of the contract.

A computer vendor would like to take the position that he will deliver and connect the various components, demonst[illegible]hat from an electrical and mechanical point of view these function, and demonstrate an operating system. At this point he would like to be paid.

As the purchaser (or lessor) of the equipment, it is necessary to insure that all of the promises made in the proposal have indeed been kept. A reasonably safe way to insure this is to indicate in the RFP the conditions for acceptance and to insist in the contract that these conditions be met. Typically, the acceptance criteria which must be met by the hardware and software delivered and installed on site should include: (a) satisfactory physical performance of the equipment, (b) satisfactory performance of the vendor supplied software, i.e. operating systems, languages, specialized programs, etc., (c) completion of the conversion, (d) demonstration that the special projects are operable, and (e) successful completion of the benchmark test.

Additional language of the following sort places the burden of compliance with all terms of the contract upon the vendor. "Upon delivery and installation of the physical equipment and software, no part may be removed by the vendor if he has successfully completed all phases of the acceptance testing unless the purchaser (lessor) defaults in payment as agreed upon." The thrust of this statement is to insure that a vendor who has had to bring in additional equipment in order to meet the acceptance criteria may not subsequently remove it. Such excess equipment should, as a minimum, be made available to the customer at no additional cost.

Another stipulation that may be added provides for uninterrupted service, even in the case of failure of the equipment to meet all acceptance criteria. "In the case that the physical equipment or the software fails to meet the acceptance tests, the equipment and software shall remain on site until that time that replacement of equipment or software from another vendor is completed. The original vendor shall be liable for all expenses, direct and indirect, that the purchaser (lessor) may incur if acquisition from a subsequent vendor becomes necessary."*[2]

III. The Evaluation of the Proposals

There will be general areas in each of which specific questions will need be asked and answers obtained and evaluated. On the technical side, the two broad categories are software and hardware; on the management side, there is the question of commitment of the vendor to remain a viable force in the marketplace.

One format for presenting the factors which will enter into the evaluation is as follows: At the top of a page, succinctly state the factor to be evaluated; example: 'Memory Access Speed'. A brief description indicative of the characteristics sought and the reasons for these should be given; for example: 'Access times to memory influence processing times and consequently throughput. Generally low access times imply short processing times and increased throughout'. Then, for each of the vendors submitting a proposal, listing a succinct comment evaluating his equipment on this point is appropriate.

It is not the intent at this level to evaluate competing computing systems but rather to indicate significant factors, remembering that these may vary depending on the particular environment in which the computer is to function. The following presents such a list:*

1. Software Pricing Policies
2. Control Program Characteristics
    - (a) Ease of Use
    - (b) Ease of Operation
    - (c) Failure Recovery
    - (d) File Security
    - (e) System Reliability
3. Programming Languages
4. Conversion Policies
5. Data Base Systems
6. Hardware - General
7. Hardware Service and Support
    - (a) Remedial Maintenance
    - (b) Preventive Maintenance
8. Disks
9. Digital-to-Analog and Analog-to-Digital
10. Communications Subsystems
11. Growth Within Product Line
12. General Hardware Characteristics
13. Peripheral Hardware Characteristics
14. Other Software Considerations
15. Benchmarks

---

* EDITOR'S NOTE: The following is a summary list of the actual paper submitted. The complete paper is available from the CAUSE National Office.

IV. The Evaluation Model

For each of the factors that was considered in previous sections, i.e. factors that were deemed significant in evaluating the proposals submitted by the vendors, assign a numerical weight which is your opinion concerning the relative significance of that factor in relation to all other factors considered.

The numerical weight to be assigned to a specific factor should corrolate directly with your estimate of the significance of that factor to your needs. Choose a convenient scale, say 0 to 100, with 100 assigned to the most important factors and smaller numbers assigned to less significant ones. For example, if the initial configuration actually requires forty (40) timesharing teletype terminals, the weight attached to this factor might be 100. While an interactive COBOL compiler may be desirable, your opinion concerning its significance to your operation may cause you to weight it only at 25. Then, for each of the factors, rank each vendor on a 0 to 1 scale insofar as that specific factor is concerned. Finally, multiply the rank by the weight and sum over all of the factors considered. That vendor which scores highest should be the desired choice.

Another measure of acceptability of a proposal is measured by the distance that the given proposal differs from an ideal. In this model, a weight is also attached to each factor. For each factor, a number which is used as a reference coordinate is arbitrarily chosen; it may as well be zero (0). Then, for the given factor and a given computer, a numerical estimate can be given which would indicate how far away from zero (0) (the ideal value) you feel the component lies. Square this number and multiply it by the weight of the factor; finally, sum over all factors. This model is a quadratic one and in a sense, measures the distance that a given computer configuration lies from the origin, or ideal. The configuration for which the distance from the ideal is smallest should be the choice.

It would be difficult at this time to try to justify either this particular linear model or the quadratic one. It would be equally difficult to try to justify any other mathematical model. Their only defense would be their ease of use and the ability to test the sensitivity of the models to different weighting factors.

The sensitivity testing in itself can be significant. If a small change in the numerical values of the weights does not produce a change in the relative ordering of the vendors, it would then be reasonable to assume that the ordering that was produced by the original model is stable and probably reflects the best choice for the installation. However, if small variations in the weights produce different ordering of the vendors then clearly the model is not very stable. In that case, one would have to go back and examine those components which are producing this shifting in relative values; further study would be needed in order to ascribe any degree of confidence in these.

If both models are sensitive to the weighting parameters, to the extent that small variations produce changes in the ordering of the vendors, then it would be reasonable to conclude that, for the factors considered, there is no really discernible difference between the proposals. The decision would then have to be based on additional factors still to be introduced, factors which could undoubtedly seem capricious. In this situation however, nothing is really lost in that if no disceribie difference can be measured, then any of the choices is acceptable.

V. The Contract

After having made the decision to acquire a specific computer, it becomes necessary to enter into the detailed contract negotiations with the vendor. At this point, one must be prepared to accept variations from the original proposal that the vendor made and to evaluate these variations in light of your own needs. On the other hand, you will have had by this time contact with other users of this particular vendor's equipment. From them you should be able to determine any special consideration that the vendor had given to them and decide if these special considerations are significant to you. If these are, you have every reason to ask for and expect to receive similar considerations although they may not have been part of your original request for proposal. Information is available from other user installations concerning those concessions that had been made to them; avail yourself of it.*[4]

It is difficult, if not impossible, to describe how one goes about negotiating with vendors over prices and conditions. Initially, of course, ask for everything you might conceivably want; some vendor may offer concessions in one area, another vendor in another area. Inquire as to the precise reasons why certain conditions can not be offered; require that the vendor's sales force go to his management in an effort to secure the terms you need. Arrangements can be arrived at in face-to-face negotiations with senior company representatives, arrangements that sales force personnel would not consider possible. These negotiations demand a sense of timing and reasonableness; they are dependent on the vendor's particular marketing strategy at the time. For example, a vendor may not be particularly interested in a sale at the time, he may be more interested in establishing a broader leasing base. Also, there are instances when it is impossible to get any concession from a vendor for circumstances which may be beyond his control, i.e. perhaps a court order prevents deviation from the standard contract.

Certain factors, however, are reasonably standard and should be mentioned. These include delivery dates, installation procedures, and responsibility for certification that the equipment is as agreed to. It is this last factor which is exceedingly important.

Under the assumption that the benchmark has been run at the vendor's installation, you can insist in the contract that the same benchmark be run on the equipment installed at your site. The purpose for doing so is to insure that the equipment upon which the benchmark was tested is not different from the equipment that you had contracted for and expect to be delivered. It is recommended that no equipment be accepted and paid for until the benchmark agreed to by both parties has actually been demonstrated to your satisfaction to have been run on your equipment. The contract should contain the stipulation that the benchmark, when run on the equipment delivered to you, performs as agreed upon or as had been demonstrated previously by the vendor; in the event that the performance falls short, the vendor will supply at no additional cost such additional hardware and/or software as may be necessary to insure compliance with this performance criterion.

As a final note, renting versus purchase should have been considered as well as the alternative of leasing from a third party. With the rapid growth of networks imminent, it is reasonable to evaluate your needs in light of the possibility of attaching to a network. The total cost for service may be less than the acquisition of the new computer and the support of personnel needed to keep it running.

*1. The thrust of this article is toward those with computer experience who are contemplating upgrading rather than those who are venturing into the field. The stakes are too high for the neophyte.

*2. (a) EDUCOM has compiled and published a list of special contractual terms that have been negotiated by colleges and universities.
"Contracting for Computing: A Checklist of Terms and Clauses For Use In Contracting With Vendors For Computing Resources." EDUCOM: Interuniversity Communications Council, Inc., P. O. Box 364, Princeton, New Jersey, July 1973.
(b) The General Services Administration of the U. S. Government annually negotiates contracts with vendors. The government is in "a most favored customer" position and as such has been able to specify contractual terms suited to its operation.

*3. Jean Sammet: Programming Languages: History and Fundamentals: Prentice Hall: Englewood, New Jersey, 1969.

*4. Each vendor works with, in some cases may provide support to, a user group. Attending one or more meetings of these groups and becoming acquainted with the individuals who participate can lead to revealing discussions.

Bibliography (Primarily a source to series of relevant materials)

1. Computer Characteristics Quarterly: Adams Associates, Inc., 128 The Great Road, Bedford, Massachusetts.

2. Computers, Control & Information Theory: National Technical Information Service, Weekly Government Abstracts: 5285 Port Roayl Road, Springfield, Virginia, 22151. See for example:

June 24, 1974 issue, Page 134: A Technical Index of Interactive Information Systems: Fife, D. W., Rankin, K., Fong, E., Walker, J. C., and Marron, B.A.

3. Federal Information Processing Standards Publication Series: U. S. Department of Commerce, National Bureau of Standards. For example: FIPS PUB 21: (C13.52:21) Common Business Oriented Language (COBOL) (Software Standard, Programming Language).

4. More on Synthetic Benchmarks: EDP Performance Review: Vol. 2, No. 5, 1974: Applied Computer Research, 8808 North Central Avenue, Phoenix, Arizona, 85020.

5. Computer Characteristics: Infosystems: a Hitchcock Publication, Wheaton, Illinois. May 1974, Vol. 21, No. 5, Pg. 39-47, 52-59.

6. Timmreck, E. M., Computer Selection Methodology. ACM Computing Surveys: Vol. 5, No. 4, December 1973, Pgs. 199-222. ACM, 1133 Avenue of Americas, New York, New York, 10036.

7. DATAPRO 70, DATAPRO RESEARCH CORPORATION, 1805 Underwood Blvd., Delran, New Jersey, 08075.

8. Avoiding Hassles with Vendors, Robert A. Bucci, Datamation, July 1974, Pgs. 68-72, Technical Publishing Company, 1301 Grove Avenue, Barrington, Illinois 60010.

MANAGEMENT APPRAISALS

SYSTEMATION BEFORE MECHANIZATION

David A. Wearley
Manager
Office of Management Analysis and Assistance
University of California
Los Angeles, California

MANAGEMENT APPRAISALS
(SYSTEMATION BEFORE MECHANIZATION)

David A. Wearley, Manager
Office of Management Analysis and Assistance
University of California
405 Hilgard Avenue, Los Angeles, California 90024

Managers and analysts must and can act together to identify operating weaknesses and thereby improve operations. Once operations are running smoothly and only when they are running smoothly, can truly effective "Management Information Systems" be constructed. A manager's job is a complex one and the management analyst's role is to help make that job less complex. The appraisal program is designed to do just that--enhance a manager's performance at any level--to help the manager to do a better job.

1. MECHANIZATION WITHOUT DIRECTION

How many of you have been involved with an EDP effort that missed its mark?

- .One that was too sophisticated and ambitious during the initial steps and had to be cut back in order to get something up and running--or
- .One that was designed merely to provide reports for higher echelons and failed to consider the operators' needs?
- .How often have you heard--"we need to mechanize that so get to it"?

2. OPPORTUNITIES FOR IMPROVEMENT

We have all had these experiences and more. I happen to believe that much of this--mechanization without direction--can be alleviated if we can agree that EDP is a means of improving the manager's lot and not an end in and of itself--EDP must be used as an opportunity for improvement and not as a status symbol.

The "opportunities for improvement" concept incorporates the pragmatic aspect of improvement as contrasted to total solution . . . an effort that seeks for the best attainable, rather than an absolute, all inclusive solution. It is a concept of movement--that considers the tradeoffs between alternatives, a divide and improve relationship rather than an all or nothing solution.

University business operations are complex. Organizational elements come in all sizes and are constantly changing, efforts and services vary considerably, and extent of operations vary. Due to these variances and changing conditions, the need for early identification of weaknesses and opportunities for improvement is paramount to success. In many universities, the status-quo prevails and management has not recognized this need. For example, some managers have become deeply concerned over weak spots in their organizational structure, administrative processes, utilization of resources, etc., but are floundering in their corrective actions. Many turn to EDP as the panacea. In order to profit by present or future opportunities, management of any university unit (regardless of size) must establish a sound means of appraising its activities. Management must determine: where there are weaknesses, where improvements are needed, what causes operations to succeed or fail. Frequently these causes relate to mechanized or as better known, Management Information Systems.

3. MANAGEMENT INFORMATION SYSTEMS

As you are aware, a Management Information System (MIS) is simply the formalized information processing system of an organization--manual and/or mechanized. Therefore any university, any campus or any organization has Management Information Systems--good, bad or indifferent though they may be.

The characteristics of these systems vary enormously, depending on the organization's size, activities, and management style. A well-designed MIS should reflect the tradeoffs that exist between independence and integration. Many development efforts have resulted in systems that are not well received by operating personnel, miss their operational target dates, are difficult to operate, have inordinate cost overruns, and require an excessive amount of redesign immediately after they go operational. A suitable balance should certainly not result in a "total" monolithic system that captures the imagination of the more fanciful members of the profession.

It is my contention that many of the conditions affecting MIS efforts are the same as those that affect manual operations and that they can be brought under control and improvements effected. All that is required is the application of a sound, systematic, open-ended approach to identification of both manual and mechanized opportunities for improvement rather than through a blind, parochial close-ended approach.

4. CRITICAL FACTORS AFFECTING UNIVERSITY IMPROVEMENT EFFORTS

A major question asked in taking on any style of improvement project is: What are the most critical factors that affect university improvement efforts? All too often the improvement effort has been one of reaction to a specific need at a specific time--responding to the pressure of the moment, often without regard to merit or to a total university systems effort--and then frequently the effort is discarded (or set aside) when a new or greater moment of crisis arises. Changes are often decided on an ad hoc basis without adequately considering how a specific decision may affect future efforts or how the resulting effort fits into what improvement efforts came before. Inadequate manual systems are constantly being mechanized that end up, far too often, as inadequate computerized systems.

Overall there is little or no "Improvement Climate" throughout most universities. Traditional thinking and parochial attitudes do exist. Most improvement actions and/or reactions run vertically up and down the organizational chain, rather than horizontally across organizational lines in consonance with the real and actual operating interactions and interfaces. Proprietary, organizational self-interests prevail, and decisions, for the most part, are still being made from that point of view--a how it affects me and my area attitude--with little attention being given to other considerations and the University as a whole. The network nature of operations, along with the mandatory considerations for economies, makes an overall integrated improvement effort an absolute.

5. MANAGEMENT APPRAISALS

A real technique for drawing on the real substance of improvement effort is the management appraisal concept--for first we must insure that operations are ready for mechanization. Weaknesses require identification early on rather than after a process has been mechanized and attempts are made to bring it up and running.

5.1 The Internal Analyst. The successful university manager of today recognizes that an effective tool to accomplish such an appraisal is available--the internal analyst (Systems, EDP, Administrative, General, etc.). In recent years the scope and magnitude of internal appraisal has greatly expanded with the arrival of the management analyst. However, in many universities the analyst is still being used in the mythical pursuit of dollar accuracy--as an auditor or to develop the mythical problem cure-all--the ideal electronic data processing system. The primary objective of a management appraisal program is to identify opportunities for improvement in any of the elements examined. The identification of opportunities for improvement is everybody's business.

5.2 Improvement Takes Time. Today's university managers all too often have neither the time nor the facilities to objectively evaluate the effectiveness of controls, the accuracy of reports, or to appraise the efficiency of their organization. In effect, management manages by crises. Thus, managers are only able to focus limited attention on high level matters: the unusual and the troublesome. It is this concept which makes the role of the internal analyst so important. The analyst bridges the gap by providing the information upon which the managers can base their decisions, by evaluating the effectiveness of business operations--not only to prevent waste of resources but to identify other unsound administrative practices--by insuring that effective lines of communication exist, and by pointing out areas requiring management attention.

5.3 Purpose of Management Appraisals. The purpose of the internal management appraisal is to provide those responsible for management at all levels (academic departments, research units, administrative activities, etc.) with a cooperative but independent, objective, and constructive evaluation of the effectiveness and efficiency with which managerial responsibilities are being carried out. Upon analysis, it is apparent that few managers take full advantage of or fully appreciate the advantages to be gained by utilizing the assistance of their own analysts in accomplishing assigned management functions. An outside consultant is accepted as a professional without question--how about the in-house consultant?

5.4 Management's Reluctance Toward Appraisals. There has been a natural reluctance on the part of university management to call upon internal analysts for assistance. To some extent, this is understandable, yet, at the same time, it indicates that management does not consider the analyst part of its team, but rather as an interloper. Too often the analyst is regarded as someone who criticizes, points out errors, tells them where they are wrong or develops grandiose EDP Systems with little regard to real need. The analyst's real contribution is assisting others in finding opportunities for improvement and helping to develop these improvements (i.e., team approach to problem solving). This attitude on management's part is unfortunate because the internal analyst exists to serve management. The best way management can be served is by objective appraisals in those areas of management interest that include recommendations for resolving noted problems rather than those containing a series of recriminations or need for an over-sophisticated Management Information System.

5.5 Internal Analyst's Need to Demonstrate Benefits. To some degree, the lack of enthusiasm on the part of management toward the management appraisal program has been generated by analysts who have not conveyed or demonstrated the benefits that can be derived from a sound and effective appraisal program designed to assist managers in carrying out their responsibilities.

To achieve optimum benefits from a management appraisal program, managers must have definite knowledge and a firm conviction that the work of the analyst is being done for their benefit--in fact supporting their own efforts.

6. COMMON OBJECTIVES FOR MANAGEMENT AND INTERNAL ANALYSTS

Internal analysts provide an additional arm for management. Augmenting its staff, they can review the day-to-day operations pointing out areas that require attention so that corrective action may be taken on a timely basis. Management and analysts must be convinced of their common objective; that is, to conduct the university's business in the most efficient, effective, and economical manner possible. Management can achieve this by establishing sound and effective policies, procedures, standards, and controls; analysts, by evaluating each factor for management and identifying opportunities for improvement.

6.1 Setting the Stage For Results. A close rapport must be established and maintained between management and analyst based on mutual respect and trust. Analysts must gain the confidence and acceptance of management at all levels by keeping faith with all levels of management. Appraisal results should be frankly discussed with the manager immediately responsible for the function before discussion with superiors, department heads, Vice Chancellors, Vice Presidents, Provosts, Chancellors, or the University President. Each unit manager should be given the opportunity to comment on any appraisal results in order to:

1. Discuss present practices,
2. Question the analyst's results, or
3. Agree with the analyst and set forth the corrective action taken and/or to be taken.

The analyst should never place any level of management in a position whereby he or she will be called upon to comment on appraisal results which have not been thoroughly discussed beforehand. The analyst must recognize and fully understand that analysts--to be professionally effective--provide helpful advice only and do not issue mandates for action.

6.2 Open Appraisals Approach. In many instances action items can and should be jointly developed by the analyst and manager. This can be accomplished by conducting an "open appraisal," i.e., by making all information and all ideas available to managers and top management at all times upon request, for review, to show that evaluations made were fair, impartial, and truly representative of existing conditions. Harsh, inflammatory, or personal language should never appear in an appraisal report and seriously detracts from the value of a project. A good conclusion, properly presented, will stand on the facts alone.

6.3 The Management Team Approach. Analysts can accomplish a great deal if management believes in and supports the management appraisal program, establishes a clear cut understanding throughout the organization of the analyst's purpose and needs, accepts the results as a constructive, independent appraisal of its operations, and considers the analyst as an integral part of the management team. Without openness and mutual understanding, difficulties will arise causing confusion and increased complications. Problem areas must be brought out into the open and jointly discussed. If they are not carefully considered, they cannot be resolved. A practical procedure is to attack problems while they are still small and easy to handle. Management is forced by the onset of the ever accelerating technological advances to be more flexible and adaptable to these rapidly changing operational conditions. Emphasis is on better ways and means that will result in improvements in the administration, support and operational aspects of the organization and in the ways they interact selectively and totally.

7. ACKNOWLEDGEMENTS

This paper represents the combined efforts of several persons in the Office of Management Analysis and Assistance and other campuses and University offices. Although it is not possible to acknowledge here, the individual contributions of everyone involved in the development, review, and preparation of this paper, their assistance was material. Without their assistance, this paper would not have been developed.

8. HANDOUTS

Two handouts were provided during the December 1974, CAUSE NATIONAL CONFERENCE in Washington, D.C., that are not included in the Proceedings. Additional copies are available upon request to the author.

8.1 David A. Wearley, Outline for Conducting a Management Appraisal Project. University of California, Los Angeles, October 1974, 9 pages.

8.2 David A. Wearley, Management Appraisals. University of California, Los Angeles, May 1974, 6 pages.

MANAGEMENT OF A DISTRIBUTED DATA BASE

B. D. Cross
Director
Management Systems
Southern Illinois University

Management of a Distributed Data Base

During the next 30-40 minutes I will be explaining a system used at Southern Illinois University at Carbondale and some of the managerial deliberations used in selecting the system. We will be talking about an old, old problem--that of registering students. I believe that you will find this solution has some noteworthy considerations for you. The application was installed in September, 1974, and as of this time has been used to register 12,000 of our 19,000 student population for the Spring Semester. Therefore, the findings of this systems applications are fresh and are being passed on to you as items hot off the press. The fact that the system was designed under a turnkey contract in which the software was provided as a part of the equipment contract may also interest you. If this software and equipment is applicable to your institution you can pick up a system fairly quickly and with little programming costs associated with it. The Sanders Data Systems Corporation looked upon this application as having wide potential usage and has created a design which is highly flexible. I realize that because of size of institution, some of the larger schools will not be able to consider this approach. I might point out that we thought that was the case also and that this has been a very important decision to us because the solution really represents an alternate plan. However, we were pleasantly surprised and the Admissions and Records office is extremely pleased with this solution.

SIU-C

Southern Illinois University is a 3-campus operation in which the computing center services the needs of the Medical School at Springfield and the Carbondale campus. Our Edwardsville campus has their own 370/135. As I mentioned earlier, our student enrollment has been around 19,000 which is down from a high of 23,000. We offer about 5,500 class offerings per semester. Our installation has as a primary support computer a 370/158 with 1 million bytes of core, 5-tapes, 2-printers, 8-3330's and uses CMS (Conversational Monitor System) operating under VM. We support about 20 instructional terminals with some RJE stations such as an 1130, 2780, etc. Even with this computer horsepower we find ourselves with limited resources in attempting to support on-line administrative applications. As with many of you, we had thought IMS and Data Base Management Concepts the way to go. However, the untimely application of budget cuts and a State moratorium on new developments caused a serious retrenching and, therefore, rethinking of our computer activities. For example, in 1972, our budget was decreased by $500,000. In 1973, we had a slight increase of $129,000 but then again in 1974 it was decreased by $71,000. We do not know for sure yet but we may be in for yet another cut next year.

My point in wading you through this see-sawing of budgets is to demonstrate the uncertain situation in which planning—very complicated planning--was to take place. It was the feeling of staff personnel in the computer center that we had lost, for the moment, opportunities in acquiring IMS, additional core, and moving into VS as well as other hardware items to support on-line activity. There was clearly a need for an alternative. The registration system was about 10 years old and had been handled by IBM 357's and in a tub file operation of about 150,000 cards. The Student Information System was well along in development except for the history record and although a proposed design has been published, but is being held up due to funding.

Personnel in the Admissions and Records office established 4 basic system requirements for the registration process:

1. Maintain a current status of enrollment and allow inquiry into that status.
2. To provide a basic class schedule and fee statement to the student upon completion of the process.
3. To collect all data for feeding the student profile module as a by product of registering.
4. To allow the alternate use of the terminal to replace key entry devices.

Detailed specifications were published by the Admissions and Records office with review support by our Management Systems staff. Specifications were completed in April and equipment bids were sought from such companies as GTE, Lockheed, Mohawk, Data Point, and Hewlett Packard--to name a few. A key element in this request for bids was that the equipment be supplied with software to operate to these specifications. Only one company agreed to meet that requirement and they did so with a corporate commitment. Evidently, they believed that sufficient colleges have this same type problem so that they were willing to create a software package with widespread potential market impact. It remains to be proven if this is true but I share this information with you so that you might benefit if at all possible.

The System's Approach

The system has several advantages, namely:

1. The equipment, consisting of 6 CRT's, disk storage units, and microprocessor appears to have many of the same advantages of a large scale on line system.

2. It was considerably easier to install. There were minimal hook up pains at the center and our software staff had only to complete a VM couple. Our applications programmers wrote a couple of programs to bridge data coming in and going back from a spool to the system.

3. The equipment was delivered two weeks prior to use and went on the air on time.

4. There is little daily impact on the central facility during the time we have major research and instruction activity.

5. There was no reprogramming of the Student Information System.

At this point I would like to take some time to explain the Sanders 810 equipment and to call your attention to the evaluation written up in DATAPRO. We considered the DATAPRO write up to be a forthright statement of the advantages and disadvantages of the equipment. In describing the Sanders 810 the following will give you a quick thumbnail sketch:

1. It is an intelligent programmable terminal.

2. It has 6 cathoray tubes.

3. It is basically an RJE station coupled to our center's 158 on a 2400 Baud dial up line.

4. It contains 8K bytes of main core.

5. There are 5 megabytes of disk, 2.5 of which are permanent and 2.5 removable.

6. The system has a 176 line per minute printer and the equipment leases for $1,583 per month as we have configured it.

The Admissions and Records office found that they could make considerable improvement in their operation by the use of this equipment. Since the major on line approach to our main center was not a viable alternative, they were willing to convert their monies expended for rental of the IBM 357's and key punches plus staff savings to fund this system. Therefore, there was no additional expense on the computer center budget which was hard hit with budget cuts. The effort took minimal computer center staff costs since the user designed the system requirements. They were especially careful to get it right the first time since program changes will cost them. There have been no modifications required beyond those made at implementation under the original contract. The department likes the control it has over the equipment. Operator training is simplified and we are able to use student operators. I would like to distribute copies of the instruction sheet which is also prepared by the equipment for the benefit of the learner operators. The equipment is geared to handle 1,000 students per day peak, considering an 8 hour work shift. The cost per student registering per year is $1.00. Although we have not had to do it yet, this system can operate for about two weeks in a stand alone mode since removable disks can be used to accrue daily transactions until repairs can be made and data forwarded to the center. Of course, no system answers all the questions. Disadvantages we see are that conflict checking between courses can not be done due to limitations of disk and core. Also, our staff is not getting the experience with on line technology and there will have to be a catching up period at the time we actually make that move. We also feel that the investment serves a limited audience whereas an investment in Information Processing and software and hardware might have allowed access and use by a wider range of users.

Overview of the Application

This system is used to register students in up to 8 courses in any semester for which there is timetable data stored on disk. A student completes a registration request form and acquires the necessary advisor, deans, etc. approvals. Reference to Exhibit I will give an overview of the student process. With this completed document, the student approaches one of the CRT work stations where the operator enters the student's name, ID number, student type and call number of the course requested. Exhibit II shows the format on the CRT. As the course being called for is keyed into the equipment a visual verification is made by the student and operator for accuracy. The status of all course offerings is contained in a master timetable file within disk. This timetable is created each semester and refreshed periodically by transmission from the computer center. Any closed or restricted classes are noted by flags on the screen and retries at other course sections can be made immediately. There is flexibility with regard to credit hour allowance, pass/fail, etc. The process of entering each course request and answering the series of questions printed on the screen is repeated for up to 8 classes. The total hours requested are summed by the equipment, a fee code is assigned, and fee amounts applied. At the completion of this step the disk files are updated with the new status of enrollment data. As the student moves beyond the CRT work station he approaches the printer where a hard copy of his transaction is available. A sample of this printout is contained in Exhibit III. In this process two things have been completed--the student registration has been completed and he has a copy of his schedule for fee payment. The equipment has captured all necessary data for the Student Information System and this is transmitted the following morning by batch to the computer center. Throughout the day the Admissions and Records office has control of the impact of registering students, taking and dropping courses, so that action can be taken to advise faculty, etc. The master timetable file is actually residing on tape in the computer center. It is updated with only those transactions that pass the very tight edits. Thus, the file in the Admissions and Records office can be refreshed periodically to keep the records synchronized. Initialization of files takes place just prior to each registration period.

In closing, let me say:

1. We are pleased with the system's performance.
2. It is not an ultimate answer but it may be optimal.
3. Given the same decision set, we would do it the same way again.

Exhibit I

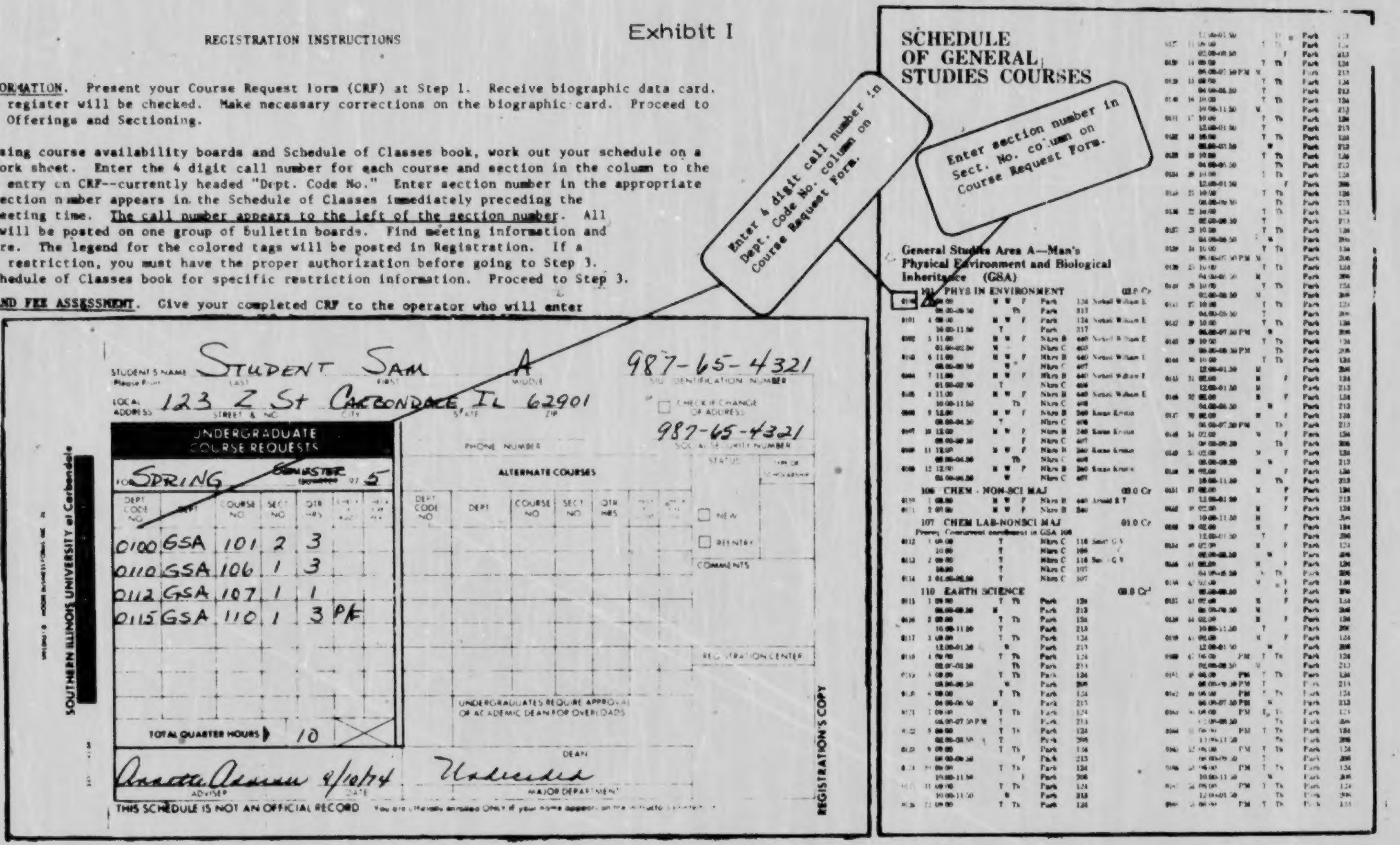
REGISTRATION INSTRUCTIONS

STEP 1. **BIOGRAPHIC INFORMATION**. Present your Course Request form (CRF) at Step 1. Receive biographic data card. Eligibility to register will be checked. Make necessary corrections on the biographic card. Proceed to Step 2--Course Offerings and Sectioning.

STEP 2. **SECTIONING**. Using course availability boards and Schedule of Classes book, work out your schedule on a registration work sheet. Enter the 4 digit call number for each course and section in the column to the left of course entry on CRF--currently headed "Dept. Code No." Enter section number in the appropriate column. The section number appears in the Schedule of Classes immediately preceding the listed class meeting time. The call number appears to the left of the section number. All added classes will be posted on one group of bulletin boards. Find meeting information and call number here. The legend for the colored tags will be posted in Registration. If a course shows a restriction, you must have the proper authorization before going to Step 3. Consult the Schedule of Classes book for specific restriction information. Proceed to Step 3.

STEP 3. **REGISTRATION AND FEE ASSESSMENT**. Give your completed CRF to the operator who will enter your registration information. Watch the screen to check the operator's accuracy and call attention to any errors. Proceed to Step 4.

STEP 4. **ENROLLMENT FORM AND FEE STATEMENT PICK-UP**. Pick up the printout of your enrollment transaction and your fee statement. Proceed to Exit.



Enter 4 digit call number in Dept. Code No. column on Course Request Form.

Enter section number in Sect. No. column on Course Request Form.

SOUTHERN ILLINOIS UNIVERSITY at Carbondale

STUDENT'S NAME Please Print: STUDENT (LAST) SAM (FIRST) A (MIDDLE)

LOCAL ADDRESS: 123 Z St (STREET & NO.) CARBONDALE (CITY), IL (STATE) 62901 (ZIP)

987-65-4321 SIU IDENTIFICATION NUMBER

☐ CHECK IF CHANGE OF ADDRESS

987-65-4321 SOCIAL SECURITY NUMBER

PHONE NUMBER

UNDERGRADUATE COURSE REQUESTS

FOR SPRING ~~SEMESTER~~ 97 5

DEPT CODE NO	DEPT	COURSE NO	SECT NO	QTR HRS		
0100	GSA	101	2	3		
0110	GSA	106	1	3		
0112	GSA	107	1	1		
0115	GSA	110	1	3	P/F	

TOTAL QUARTER HOURS ▶ 10

ALTERNATE COURSES

DEPT CODE NO	DEPT	COURSE NO	SECT NO	QTR HRS		

STATUS

☐ NEW

☐ REENTRY

COMMENTS

REGISTRATION CENTER

UNDERGRADUATES REQUIRE APPROVAL OF ACADEMIC DEAN FOR OVERLOADS

DEAN

ADVISER: Annette Adams DATE: 9/10/74 MAJOR DEPARTMENT: Undecided

THIS SCHEDULE IS NOT AN OFFICIAL RECORD

REGISTRATION'S COPY

SCHEDULE OF GENERAL STUDIES COURSES

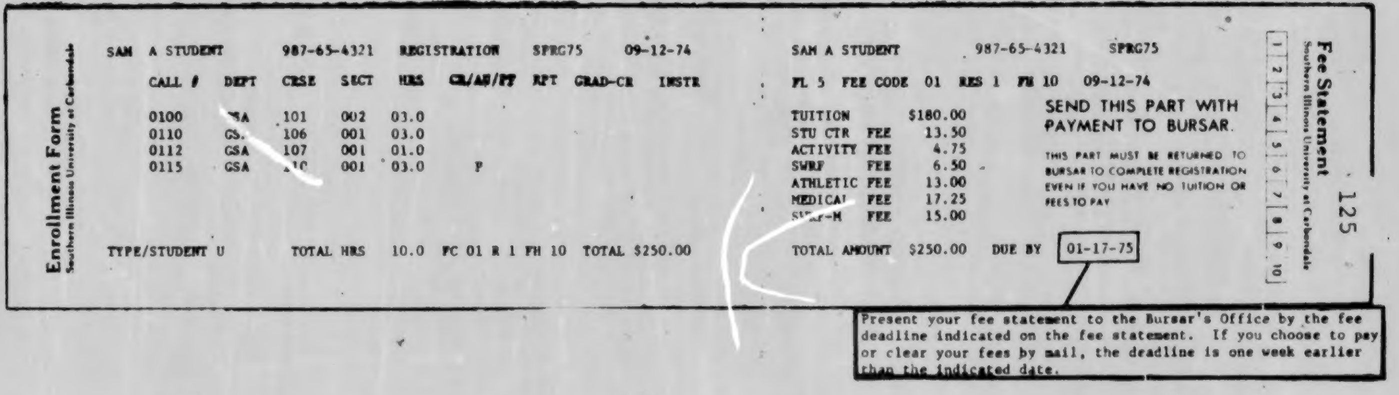
General Studies Area A—Man's Physical Environment and Biological Inheritance (GSA)

101 PHYS IN ENVIRONMENT

106 CHEM - NON-SCI MAJ

107 CHEM LAB-NONSCI MAJ

110 EARTH SCIENCE



Enrollment Form
Southern Illinois University at Carbondale

```
SAM  A STUDENT       987-65-4321   REGISTRATION   SPRG75     09-12-74

     CALL #   DEPT   CRSE   SECT   HRS   CR/AU/PF  RPT  GRAD-CR  INSTR

     0100     GSA    101    002    03.0
     0110     GS.    106    001    03.0
     0112     GSA    107    001    01.0
     0115     GSA    110    001    03.0       P

TYPE/STUDENT U      TOTAL HRS    10.0  FC 01 R 1 FH 10  TOTAL $250.00
```

Fee Statement
Southern Illinois University at Carbondale

1 2 3 4 5 6 7 8 9 10

```
SAM A STUDENT         987-65-4321     SPRG75

FL 5  FEE CODE  01  RES 1  FH 10   09-12-74

TUITION          $180.00
STU CTR   FEE      13.50
ACTIVITY  FEE       4.75
SWRF      FEE       6.50
ATHLETIC  FEE      13.00
MEDICAL   FEE      17.25
SWRF-M    FEE      15.00

TOTAL AMOUNT     $250.00     DUE BY  01-17-75
```

SEND THIS PART WITH PAYMENT TO BURSAR.

THIS PART MUST BE RETURNED TO BURSAR TO COMPLETE REGISTRATION EVEN IF YOU HAVE NO TUITION OR FEES TO PAY

Present your fee statement to the Bursar's Office by the fee deadline indicated on the fee statement. If you choose to pay or clear your fees by mail, the deadline is one week earlier than the indicated date.

Exhibit II

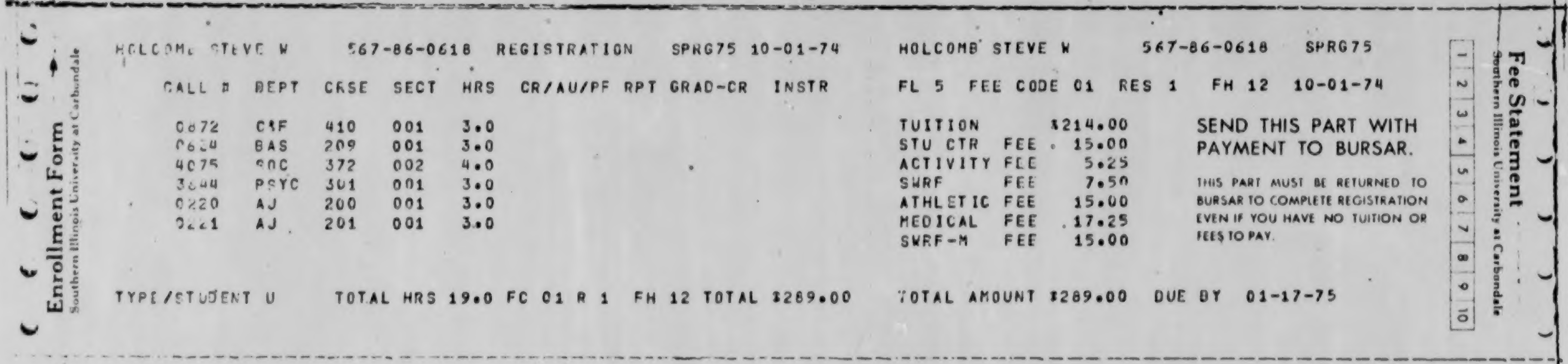
SCREEN FORMAT FOR ORIGINAL REGISTRATION

```
01 NAME -------------------  ID --- -- ----  TYPE -  TERM SPR075   DATE 09/23/74
 CALL NO. DEPT CRSE SEC HRS  C/A/PF RPT G-CR    INST    M.O.    MAX ENRL   MESSAGE
  ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
  ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
  ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
  ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
  ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
  ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
  ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
  ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
TOT HRS --.- FL - FC -- R -  PH --          TOTAL AMOUNT $---.-- DUE BY 01/17/75
```

SCREEN FORMAT FOR PROGRAM CHANGE

```
02 NAME ------------------  ID --- -- ----  TYPE -  TERM SPR075   DATE 09/23/74
 CALL NO. DEPT CRSE SEC HRS  C/A/PF RPT G-CR    INST    M.O.    MAX ENRL   MESSAGE
A ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
D ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
D ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
  ----    ---- ---- --- --.-   -     -    -      --      -      --- ---, ----------
 CALL NO. DEPT CRSE SEC HRS  WK/WD
D ----    ---- ---- --- --.-  --
R ----    ---- ---- --- --.-  --
O ----    ---- ---- --- --.-  --
P ----    ---- ---- --- --.-  --
```

Exhibit III



Enrollment Form
Southern Illinois University at Carbondale

HOLCOMB STEVE W   567-86-0618   REGISTRATION   SPRG75 10-01-74

CALL #	DEPT	CRSE	SECT	HRS	CR/AU/PF	RPT	GRAD-CR	INSTR
0872	C&F	410	001	3.0				
0634	BAS	209	001	3.0				
4075	SOC	372	002	4.0				
3844	PSYC	301	001	3.0				
0220	AJ	200	001	3.0				
0221	AJ	201	001	3.0				

TYPE/STUDENT U   TOTAL HRS 19.0 FC 01 R 1 FH 12 TOTAL $289.00

Fee Statement
Southern Illinois University at Carbondale

1	2	3	4	5	6	7	8	9	10

HOLCOMB STEVE W   567-86-0618   SPRG75

FL 5 FEE CODE 01 RES 1 FH 12 10-01-74

TUITION		$214.00
STU CTR	FEE	15.00
ACTIVITY	FEE	5.25
SWRF	FEE	7.50
ATHLETIC	FEE	15.00
MEDICAL	FEE	17.25
SWRF-M	FEE	15.00

SEND THIS PART WITH PAYMENT TO BURSAR.

THIS PART MUST BE RETURNED TO BURSAR TO COMPLETE REGISTRATION EVEN IF YOU HAVE NO TUITION OR FEES TO PAY.

TOTAL AMOUNT $289.00 DUE BY 01-17-75

MANAGEMENT SYSTEMS IN HIGHER EDUCATION

Dr. Warren W. Gulko
Director
Budget and Institutional Studies
University of Massachusetts

MANAGEMENT SYSTEMS IN HIGHER EDUCATION

Dr. Gulko discussed the pre-conditions for effective management in higher education. He also discussed the components of management systems in higher education. Also included in his presentation was the role of institutional research, management information systems, planning and budgeting.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

DECISION MAKING IN THE EARLY STAGES OF DEVELOPING A NEW ADMINISTRATIVE SYSTEM (I.E. TOOLING UP A NEW PROJECT FROM SCRATCH)

Dr. Wayne Muth
Director
Computer Center
West Virginia University

DECISION MAKING IN THE EARLY STAGES OF DEVELOPING A NEW ADMINISTRATIVE SYSTEM (I.E. TOOLING UP A NEW PROJECT FROM SCRATCH)

The first casual mention of some new need; getting an audience for the guy who thought up the bright idea; first cut at "analyzing" what might be involved; the "first design meeting"; trying to decide whether to push ahead some more (who decides?); some kind of a reasonably decent feasibility study; recalibration: "where do we stand thus far"; pushing ahead some more; who will, can, should, decide to push on or not; what are the implications of devoting resources to this new venture; the "formal charge to push ahead"; getting the actual analysis process underway--what troops? when? how? who's in charge?; investigation of alternatives for utilizing the available manpower--how can we man this project in a manner which is compatible with the University's "organization chart"--might there be a "better organizational structure";...; putting the new system into use.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

THE QUESTION OF A SYNTHESIS IN HIGHER EDUCATION MANAGEMENT

Bernard S. Sheehan
Director
Office of Institutional Research
The University of Calgary

THE QUESTION OF A SYNTHESIS IN HIGHER EDUCATION MANAGEMENT

Bernard S. Sheehan
Office of Institutional Research
The University of Calgary

The number of topics in higher education management may be finite, but they don't seem countable. Thus, while one can have a good appreciation of many of these notions, most analysts and administrators are left to bridge the gaps themselves--to provide their own synthesis. The purpose of this paper is to make tentative suggestions on the relation among certain notions often discussed under some aspects of postsecondary education management. This is not a scientific treatise, although a secondary purpose is to encourage discussion of what might be called higher education management science.

The Need and Value of a Synthesis

Information is a complex notion. It is difficult to talk about information without betraying one's own perspective of the value of different categories of facts, of decision-making regimes or styles, one's notion of the purpose or philosophy of education, of management, of government, of society, and so on. Institutions require information from professors, students and a multitude of other sources to assist decision making at many levels. The professor, as teacher, as researcher, and as expert contributor to several constituencies, needs information on which to base decisions he or she must make alone, or as part of some collective. Similarly, students as learners and as participants in all aspects of institutional affairs, need information to support judgements. Information for decisions is useful only in a context and when it is possessed along with adequate collateral knowledge of the context, that is, the process, the institution, the system, or the situation. Information is currency. The more fully it is exchanged, that is, communicated and understood, the higher the probability it will influence decisions.

The synthesis we want to achieve must give a way to structure information, which makes it easy to use for decisions in higher education management. Since most practitioners of higher education management were trained in other disciplines, they look for the system, the order, the fundamental principles, which give guidance and insight into day-to-day problems. No one is naive enough to expect an operations manual, but there must be something beyond political agility. Hills[1] has suggested the following minimum requirements for science of organization:

(1) the selection of the basic elements or units of organization, both in the context of structure and process;

(2) the invention of categorical systems or descriptive frames of reference, in terms of which the character and relational properties of units are described, or in terms of which facts about the units are stated;

(3) sufficient assessment to indicate which of the possible states of the system are empirically observable, i.e., the discovery of the laws holding among the variables in terms of which units are characterized; and

(4) the invention of a theoretical account from which both the lawful relations already observed, and hypotheses concerning additional relations can be derived.

Thus, if it exists, the science of higher education as organization would be a major aspect of the synthesis we seek.[2]

Application of the systems approach to management, even to higher education management, is not new. Opinion is divided on the usefulness, limitations, consequences and dangers of this approach.[3] No one wants to kill the goose that lays the golden eggs, either by the mindless application of simplistic analytical constructs, or by a vacuum of leadership on campus, or by suffocating system-wide or government controls, or by any other means. However, there are many people--students, professors, administrators, citizens--who want to have a better understanding of higher education and its management. In higher education, we should not avoid the basic intellectual drive to "know thyself," and your environment.

---

[1] R. Jean Hills, Toward a Science of Organization (Eugene, Oregon: University of Oregon Press, 1968), p. 15.

[2] See also, Talcott Parsons and Gerald M. Platt, The American University (Cambridge, Massachusetts: The Harvard University Press, 1973).

[3] Earl F. Cheit, "The Management Systems Challenge: How to be Academic Though Systematic," paper presented to 56th Annual Meeting of American Council on Education, Washington, D. C., October 11, 1973.

Perhaps any synthesis which is less than perfect will exclude important issues and insights. Yet the search itself forces one to question values, procedures, processes, structures and measures within a framework, systematically. Research should not proceed by rationalistic methods only, but with continued reference to the empirical meaning of elements. In any case, for the sake of higher education and the preservation of traditional values, a comprehensible synthesis is necessary now. Surely the several solitudes, but particularly faculty and administrators, should cooperate in evolving a synthesis, or many, which will assist proper, legitimate, fair and intellectually honest management of institutions and postsecondary systems.

The Essential Complexity of Higher Education

Universities share the intangibles of learning with one-room schools and the complexity of organization with business corporations. Higher education institutional and system management does not differ generically from that of any enterprise. What sets institutional management apart from other organizations is the difficulty of quantifying, even conceptualizing, educational processes. Further differences arise because of organizational structures, university traditions of consensus-seeking decision making, academic freedom and institutional autonomy, and the coordination of institutions in multilevel systems to meet regional, state and national needs. Yet in spite of complexities, all activities must be managed to meet the multifarious goals and objectives of the collective, the institution and its many components. Management decisions must be formulated on processes for which measurement of inputs, outputs, efficiency and benefits have difficult application.

Elements of One Synthesis -- Higher Education Management Space

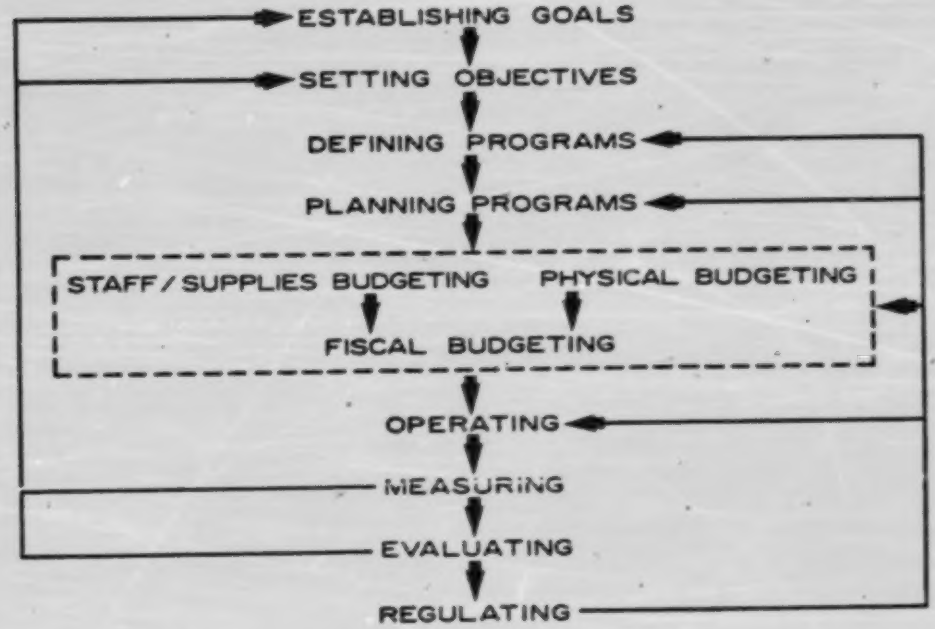
There are a number of notions and ordered concepts which, within some sphere of higher education management activity, can provide a focus for integration. Visualize a three dimensional space, call it "higher education management space." In your mind's eye, label the three axes "management functions," "program classification structure," and "organization structure"; so that this space or cube is a three-dimensional matrix within which higher education management may be contained, analyzed and discussed. The notion of orthogonal axes may be questionable, but the idea of a space is intuitively appealing and stimulates sufficient imagery to assist us in generating further ideas toward a possible synthesis. Each of the proposed axes of higher education management space is a rich source of structure and language, but each is complex and nonunique.

Management Functions

Figure 1, "One View of Higher Education Management Functions," casts management functions into a unifying framework which, while overly simplified, demonstrates interrelations and permits the process to be viewed as a continuum. Higher education management is not as much a science as an art. One may make it appear unduly scientific (almost programmable) by positing structures such as the one given the figure. This is not the intent, and once having acknowledged the danger of oversimplification, one must not become paralyzed by it. The statement of institutional goals is a qualitative description of the institution's purpose, role, and major missions. This statement articulates the several results or achievements to which the institution aspires, and to the attainment of which it will apply its means and capabilities. Objectives state institutional goals in more specific qualitative and quantitative terms. The management function of defining programs assembles entities which synergetically perform institutional missions in a prescribed way. Program planning generates specific information for resource requirements needed to formulate staff and supplies, physical and fiscal budgets.

Figure 1

ONE VIEW OF
HIGHER EDUCATION MANAGEMENT FUNCTIONS



Physical budgeting begins with facilities planning, which involves determining for the future the amounts of each type of space and other facilities required. The difference between inventory and requirements gives essential information for the facilities development program, determination of priorities, and estimated cost of physical facilities for succeeding annual budget periods.[1] Staff and supplies budgeting is the dual of physical budgeting; instead of capital resources, it relates to the other resources which must be budgeted. The management function of fiscal budgeting includes the tasks which link planning functions with operating or program execution. The normal fiduciary budget expresses the program plan for a given fiscal year in terms of dollars per chart of account line-items, including organizational units, and is used as an instrument of internal fiscal control

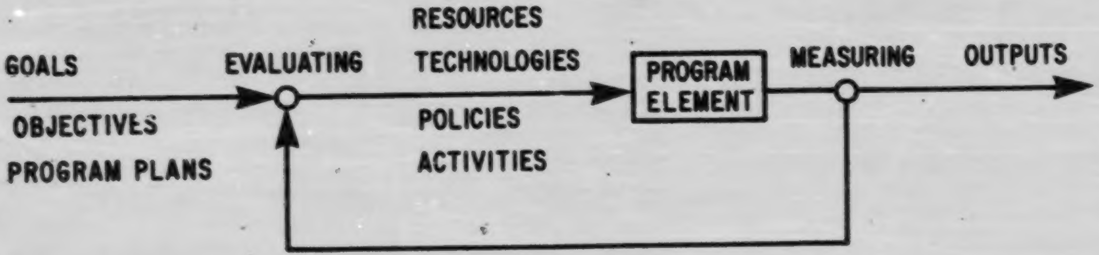
---

[1] Harold L. Dahnke, et al., Higher Education Facilities Planning and Management: An Overview, Technical Report 17-1 (Boulder, Colorado: Western Interstate Commission for Higher Education, 1971).

Operating means executing primary programs of instruction, research and public service, as well as the necessary support programs. Measuring is the management function of determining, assessing, and calibrating institutional outputs in qualitative and quantitative terms. Evaluation, judgement on the worth of something, is based on comparison of goals and objectives with outputs. Ideally, goals are outputs. Practically, some outputs differ from goals. This difference is management information, which, when evaluated, may lead to decisions for appropriate modifications designed to reduce differences. Regulation is that management control function by which the institution assures that operations stay within budgeted limitations and, insofar as feasible, that actions taken by members are in unison with institutional plans and policies.

Figure 2

PROGRAM ELEMENT VIEW OF MANAGEMENT FUNCTIONS



Program Classification Structure

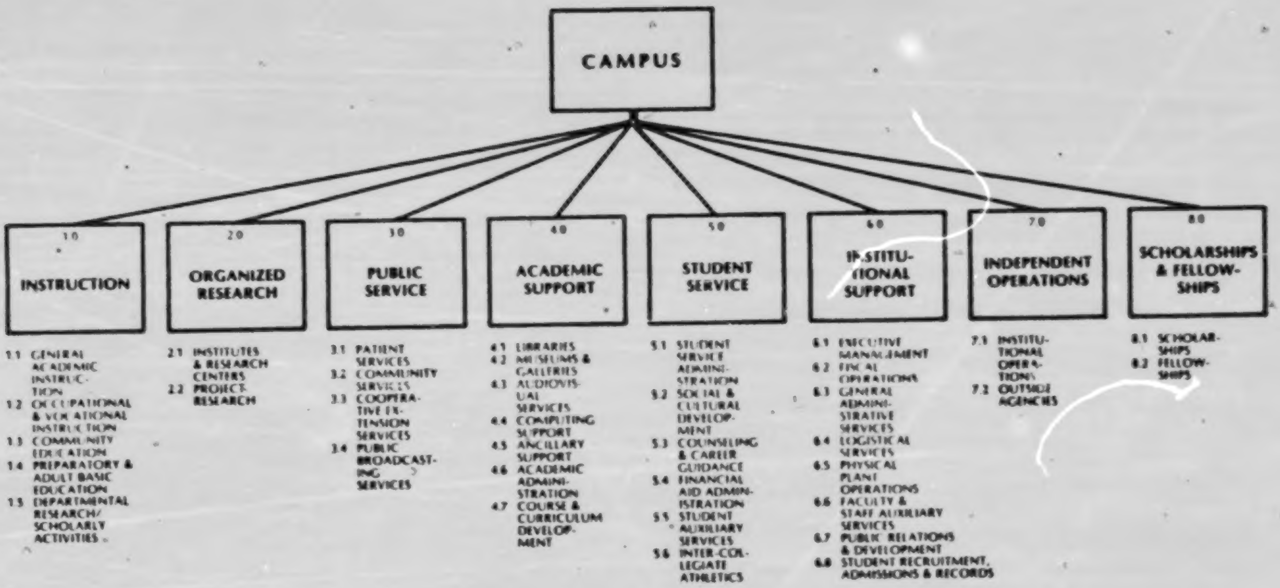
Our second proposed axis in higher education management space is a program classification structure which provides a consistent means of identifying and organizing activities which may be defined at the program level of aggregation. An example of a program element is an instructional course section. The course is thought of as an operation which produces outputs (provides knowledge and learning) by integrating or uniting a unique set of resources, technologies, policies and activities. Program element outputs are measured and evaluated against goals, objectives and program plans as illustrated in Figure 2, "Program Element View of Management Functions."

The black box diagram is useful for present purposes because it collapses numerous detailed items into the program element and permits one to concentrate on the general notions of management functions and programs within the context of other aspects of higher education management.

A program is a set of program elements serving a common set of objectives. The assignment of program elements to programs (as in a classification structure) is based on the "primary intent" of the program element. Figure 3 shows the latest version of the NCHEMS Program Classification Structure.

Figure 3

PROGRAM CLASSIFICATION STRUCTURE[1]


CAMPUS
10
INSTRUCTION
20
ORGANIZED RESEARCH
30
PUBLIC SERVICE
40
ACADEMIC SUPPORT
50
STUDENT SERVICE
60
INSTITUTIONAL SUPPORT
70
INDEPENDENT OPERATIONS
80
SCHOLARSHIPS & FELLOWSHIPS
1.1 GENERAL ACADEMIC INSTRUCTION
1.2 OCCUPATIONAL & VOCATIONAL INSTRUCTION
1.3 COMMUNITY EDUCATION
1.4 PREPARATORY & ADULT BASIC EDUCATION
1.5 DEPARTMENTAL RESEARCH/ SCHOLARLY ACTIVITIES
2.1 INSTITUTES & RESEARCH CENTERS
2.2 PROJECT RESEARCH
3.1 PATIENT SERVICES
3.2 COMMUNITY SERVICES
3.3 COOPERATIVE EXTENSION SERVICES
3.4 PUBLIC BROADCASTING SERVICES
4.1 LIBRARIES
4.2 MUSEUMS & GALLERIES
4.3 AUDIOVISUAL SERVICES
4.4 COMPUTING SUPPORT
4.5 ANCILLARY SUPPORT
4.6 ACADEMIC ADMINISTRATION
4.7 COURSE & CURRICULUM DEVELOPMENT
5.1 STUDENT SERVICE ADMINISTRATION
5.2 SOCIAL & CULTURAL DEVELOPMENT
5.3 COUNSELING & CAREER GUIDANCE
5.4 FINANCIAL AID ADMINISTRATION
5.5 STUDENT AUXILIARY SERVICES
5.6 INTER-COLLEGIATE ATHLETICS
6.1 EXECUTIVE MANAGEMENT
6.2 FISCAL OPERATIONS
6.3 GENERAL ADMINISTRATIVE SERVICES
6.4 LOGISTICAL SERVICES
6.5 PHYSICAL PLANT OPERATIONS
6.6 FACULTY & STAFF AUXILIARY SERVICES
6.7 PUBLIC RELATIONS & DEVELOPMENT
6.8 STUDENT RECRUITMENT, ADMISSIONS & RECORDS
7.1 INSTITUTIONAL OPERATIONS
7.2 OUTSIDE AGENCIES
8.1 SCHOLARSHIPS
8.2 FELLOWSHIPS

---

[1] "Revised Program Classification Structure," National Center for Higher Education Management Systems at WICHE Newsletter (December, 1974).

A program element is a discrete management unit which represents the smallest, unique collection of resources that are output-producing activities; that is, the set of resources and activities integrated so that no one activity or resource, without the contribution or support of the others, can contribute to the objective in a measurable fashion.[1] Program measures characterize each program element by indicating the resource utilization and the output associated with activities of the program element. Program measures are information items which quantify the human and physical resources assigned or utilized, activity levels generated, groups or clients served, expenditures incurred, who will bear the costs, revenue generated and outcomes expected by program element.[2]

Organization Structure

The above presentation of higher education management functions may convey the impression that the abstracted process occurs in a simple, definable organization structure ordered to meet system and institutional goals. Such is not the case. It is argued[3] that institutional structures are not relevant to current programs of research and are antithetical to public service programs directed toward the benefit of the community. Since organization structures are both variable and familiar, we will not pursue the topic, except to note the following. Organization structure is a proposed axis in higher education management space; therefore, we take the view that each management problem or situation has an organizational dimension or perspective.

---

[1] Warren W. Gulko, Program Classification Structure, Technical Report 27 (Boulder, Colorado: Western Interstate Commission for Higher Education, 1972).

[2] James R. Topping and Glenn K. Miyataki, Program Measures, Technical Report 35 (Boulder, Colorado: Western Interstate Commission for Higher Education, 1973).

[3] James A. Perkins, ed., The University as an Organization, A Report for the Carnegie Commission on Higher Education (New York: McGraw-Hill, 1973).

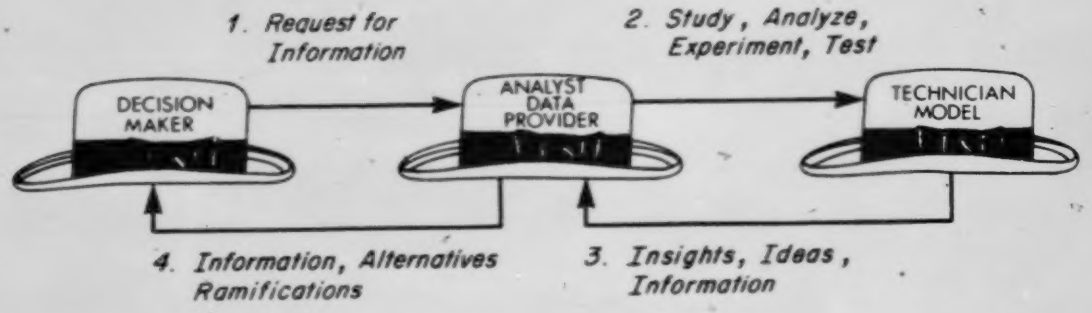
Situation Analysis

It might appear that "analysis" is out of place in a discussion on synthesis, since analysis is the separation of something into its parts, so that the parts or elements may be examined. Also, analysis refers to the resultant statement that describes the outcome of this examination. This understanding of analysis recognizes that higher education management may be separated into many sets of components or elements, depending on the purpose of the critical examination and the resultant statement. Since our purpose relates to management and higher education, we consider the breakdown of both management and higher education into elements for the purpose of associating with the elements information useful for decision making. We conjecture that insight to synthesis through repeated analysis may be helpful here, as it is in other disciplines.

The synthesis we pursued is one which evolves from the point of view of the analyst required to provide information for decision making. It relates to the role of the analyst or data provider as illustrated in Figure 4, "The Three Hat Theory."

Figure 4

THE THREE HAT THEORY


1. Request for Information
2. Study, Analyze, Experiment, Test
DECISION MAKER
ANALYST DATA PROVIDER
TECHNICIAN MODEL
4. Information, Alternatives Ramifications
3. Insights, Ideas, Information

In our office, we developed the "Three Hat Theory," which says that in responding to the need for management information, analysts must be sufficiently versatile to assume the perspectives of three people: first, the person asking for information and choosing to use it for decision making, say, the president; secondly, the analyst, wearing his own hat and translating the information needed into terms which will admit a solution by those means available, that is, taking into account imprecisions of the question, limitations of the data base, of available tools and techniques, time, talent, and other resources for proper analysis; and finally, the technician to whom the technical aspects of gathering information are clear and the meaning of the resultant data is unmistaken. Thus, data providers must live in a synthesized or integrated intellectual environment to be a successful and complete analyst. Sometimes the three characters in the above sketch are referred to as members of the management team. Ideally, each member of the team should have the potential of wearing the other two hats, thus effecting a division of labor and a multiplication of comprehension.

In the normal systems approach, one considers that a system is the synthesis or interconnection of subsystems, which have logically independent existence, function, purpose and meaning within the larger system. However, faced with a particular problem or set of problems the analyst may choose to break down the system in a way peculiarly meaningful to the situation under review. The elementary model discussed next grew out of this sort of rationale. Practice with repeated situation analysis seems helpful in evolving a larger and larger synthesis from structured practical experience.

The Elementary Model--Another Synthesis

Another synthesis is achieved by an analytical construct which gives the relationship among a number of fundamental variables. In the following example, the primary purpose is to suggest a conceptual way to pull together a lot of ideas. However, the elementary model which emerges has practical potential and has been used to assist the budgeting process at one university for several years.

The elementary model shows relationships between the number and distribution of academic staff, the number and distribution of students, and certain variables which quantitatively indicate academic and fiscal policy. It is difficult to separate academic and fiscal policy variables. For example, the number of lectures a professor gives each week influences the cost per lecture, but also has academic ramifications. The more time in the classroom or class laboratory, the less time the professor has for research, public service and administrative activites. For the moment, we are concentrating on the professor as teacher.

The elementary model combines the following well-known relations. The number of students enrolled in all course sections divided by the average section size yields the number of sections. Thus, we have the following simple relationship.

$$\frac{a}{b} = s$$

where:
a = the number of section enrolees
b = the average section size
s = the number of sections

If we multiply the number of sections by the average number of hours per week that sections meet "c", we have the number of faculty contact hours or sections hours "h". Thus,

$$a \cdot \frac{c}{b} = h$$

Now, if we divide the number of section hours "h" by the average teaching load "d", that is, the average number of section hours per full-time equivalent professor, we get the number of full-time equivalent professors (FTE) required to provide the program described by the variables "a", "b", "c". Thus, the elementary model is

$$FTE = a \cdot \frac{1}{b} \cdot c \cdot \frac{1}{d}$$

The form of the elementary model chosen produces an explicit relationship among five of the most fundamental and commonly used variables in analysis in higher education management. The elementary model links the distribution of students in course activities with the distribution of academic teaching staff needed to sustain these activities within institutional policy. Matching these two distributions is one of the basic institutional management problems. The other three variables may be considered policy variables since their numerical values may be chosen to reflect certain institutional policies. The average section size "b" relates both to academic policy and fiscal policy. The average section size may vary among disciplines, instruction types (classroom, class laboratory, tutorial, etc.), and levels of instruction (freshman, graduate, etc.). The average section size is perhaps the most sensitive variable in determing the cost of instruction.

The average section hours per section "c" is likewise an academic and fiscal variable. Length of time a section should meet per week for effective teaching is an academic matter, but this duration is also an important factor in instructional cost. The number of section hours per FTE faculty "d" is another policy variable with academic and fiscal consequences associated with each value chosen. Depending on the situation and problem under review, any, all, or none of the five may be considered policy variables. For example, an enrolment ceiling would fix "a" section enrolees and a hiring freeze would set FTE staff; thus, under these circumstances, "a" and FTE are policy variables.

If we wish to examine the elementary model in more detail, it is possible to introduce explicitly more policy variables. For example, the instruction-type mix, that is, whether a department teaches its courses as lectures, tutorials, laboratories, etc., is another important decision which is based on academic and fiscal policy. If we define "g" as the ratio of section enrolees to course enrolees, then the variable "a", section enrolees, may be replaced by substitution of the following expression:

$a = a_c \cdot g$ where a_c = *course enrolees*

We can carry this unraveling further by defining another policy variable "f" as follows:

$a_c = a_s \cdot f$ where a_s = *the number of students*
f = *number of course enrolees per student*

To build the synthesis further, replace the policy variable "f" with a matrix whose elements have the dimensions of course enrolees in discipline "i" per student registration in program "j". This substitution permits association of student enrolment projections by student academic program to the consequential distribution of section enrolees required. The matrix is just a slight variation in units on the induced course load matrix (ICLM).[1]

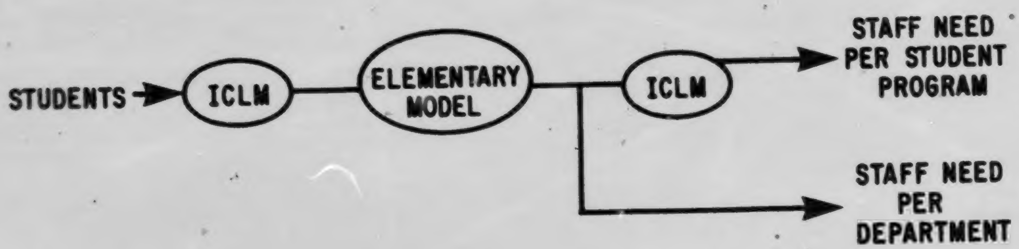
---

[1] Michael J. Haight and Charles W. Manning, Induced Course Load Matrix Generator, System Documentation (Boulder, Colorado: National Center for Higher Education Management Systems at the Western Interstate Commission for Higher Education, August, 1972).

Thus we have a model to this point which calculates academic staff need requirements by department given student enrolment figures. Since we have the ICLM, we may also relate the staff need back to each student academic program.[1] The components of the model are shown in Figure 5, "Augmented Elementary Model."

Figure 5

AUGMENTED ELEMENTARY MODEL



It may be argued that the elementary model is limited because it relates only to the "coursified" component of the instruction program. A case in point is the supervision of graduate students. Since the staff requirements vary as the number of graduate students supervised, the following may be added to the elementary model:

$k \cdot \frac{e}{d}$. where: k = *conversion factor relating graduate students to section hour equivalents*
e = *number of graduate students*

---

[1] The logic of this model parallels the Resource Requirements Prediction Model (RRPM) of the National Center for Higher Education Management Systems (NCHEMS). K. M. Hussain, Resource Requirements Prediction Model (RRPM-1): Guide for the Project Manager, Technical Report 20 (Boulder, Colorado: National Center for Higher Education Management Systems at the Western Interstate Commisssion for Higher Education, 1971). For more comparative details and applications, see: Bernard S. Sheehan and Mervin G. Michaels, "Academic Program Analysis for Budgeting," Public Policy: Issues and Analyses, Ed. Robert G. Cope (The Association for Institutional Research, 1974); and A. J. M. Tebbutt, "Conversion of RRPM to Accommodate AREG," Office of Institutional Research Internal Report 104i (Calgary: The University of Calgary, 1974).

Thus, the elementary model in the following form

$$FTE = a_s \cdot f \cdot g \cdot \frac{1}{b} \cdot c \cdot \frac{1}{d} + k \cdot e \cdot \frac{1}{d}$$

is an example of a useful analytical construct, which pulls together a number of related ideas. Studying the potential and limitations of variations on the theme struck by this proposed version of the elementary model is a productive exercise. One can show that the elementary model is central to a number of practical and widely used analytical constructs. Perhaps more germane to our current discussions is that familiarity gained by the study of the elementary model helps one develop an intuitive understanding of the important dynamics of higher education and provides a rational and structured way of raising important further questions for examination.

Application of the elementary model as an aid to institutional budgeting takes the synthesizing process into the practical arena. Since staff requirements are perhaps the single most important item in any departmental budget, a given academic staff distribution turns out in practice to be a useful starting point in institutional budget discussions.

The staff need distribution generated by the model does not take into account variations among departments in staff needs for research and public service. Also, the model distribution may be used as a guideline to operating budget distribution at the school or college level--not the departmental level. The actual budget distribution at the departmental level involves a decision-making process of a more traditional sort and one which will be discussed in the last section.

The budget discussions and investigation which follow a comparison of the elementary model academic staff distribution, the actual departmental staff distribution and the departmental proposals for staff are further aids in synthesis. The institution may evolve a better appreciation of the interdependence of data bases that must be maintained for it to provide information to the elementary model; terminology surrounding the elementary model may clarify communication among departments and those responsible for budgeting at several levels; and variables which are either input or policy variables to the elementary model may become standard data and readily available for other decision-making purposes.

Fuzzy Sets

In many disciplines it is customary to assume that imprecision in decision processes is equivalent to randomness and can be dealt with using the concepts of probability theory and the tools of decision theory, control theory, and information theory. Bellman and Zadeh[1] noted the need to differentiate between randomness and fuzziness, with fuzziness being a major source of imprecision in decision making. Zadeh uses fuzziness to mean the type of imprecision associated with fuzzy sets, where fuzzy sets are that class of sets in which there is no sharp transition from membership to nonmembership.[2]

A new approach to analysis of complex systems and decision processes, which uses linguistic variables as well as numerical variables, conditional statements among fuzzy variables, as well as fuzzy algorithms, has been suggested.[3] Zadeh notes that the ability to manipulate fuzzy sets and the consequent summarizing capability constitute one of the most important assets of the human mind, and one that distinguishes human intelligence from the machine intelligence embodied in present-day computers. He goes on,

> Viewed in this perspective, the traditional techniques of system analysis are not well-suited for dealing with humanistic systems because they fail to come to grips with the reality of the fuzziness of human thinking and behavior. Thus, to deal with such systems realistically, we need approaches which do not make a fetish of precision, rigor, and mathematical formalism, and which employ instead a methodological framework which is tolerant of imprecision and partial truths.[4]

---

[1] R. E. Bellman and L. A. Zadeh, "Decision-Making in a Fuzzy Environment," Management Science, Vol. 17, No. 4 (December, 1970), pp. B141-B164.

[2] L. A. Zadeh, "Fuzzy Sets," Information and Control, Vol. 8 (June, 1965), pp. 338-353.

[3] L. A. Zadeh, Outline of a New Approach to the Analysis of Complex Systems and Decision Processes, Electronics Research Laboratory Memorandum No. ERL-M342 (Berkeley: University of California, 1972).

[4] Ibid., p. 4.

The application of fuzzy sets seems to differ from the simulation modeling approach to a synthesis in a fundamental way. Normal simulation modeling proceeds from the real situation to an abstraction to a computer realization of the modeled abstraction. Typically, the simulation of the abstraction is exact or at least limitations are well defined because the characteristics of the realization and the abstraction are usually given by detailed quantitative specifications. Imprecision in simulation modeling is introduced at the stage of developing the abstraction.

In the fuzzy set approach, it appears that the ultimate model can be manipulated and described with the same rigor as in normal simulation modeling. The essential difference seems to be the way the imprecision in the abstraction is handled. It is not simply neglected (or to be taken into account "outside" the model later), but is recognized by the definition of the variables and the relation among the variables in the abstraction. While it is too soon to gauge the impact fuzzy sets will have on the evolution of a new synthesis in higher education, it is clear that the first steps are promising.

Synthesis Unraveling?--Up the Down Paradigm

An editorial on the energy crisis in a recent issue of Science, entitled "On Incrementing a Hurricane" by Amitai Etzioni, went as follows:

> Scientific studies of national decision-making have shown that democracies tend to abhor comprehensive planning and major shifts in strategy, preferring instead to "increment" or muddle through, making adjustments through a series of small steps aimed at moving away from a mess rather than at moving toward clearly stated, positive goals. . . . Muddling through, the incrementalists conclude, is not just a fact of life, but the best way to live. . . . While incrementing might well be the genius of democracy under most circumstances, to try to muddle through a hurricane is folly. . . . this crisis . . . will require more master planning and national coordination than we tend to welcome in peacetime. It will require more leadership and less incrementation.[1]

---

[1] Amitai Etzioni, "On Incrementing a Hurricane," Science, Vol. 186, No. 4166 (November 29, 1974), p. 783.

These two competing paradigms of decision processes, comprehensive vs. incremental, are discussed in the higher education context by Frank A. Schmidtlein.[1] The first conception, "rational/comprehensive/prescriptive" (RCP), is associated with the notions of modern management tools and techniques, planning program budgeting systems, systems analysis, decision analysis, management information systems, etc. It has been argued that the growing influence of the work of the National Center for Higher Education Management Systems (NCHEMS) attests to the increasing acceptability and value of the RCP approach. The competing "successive comparisions/incremental/remedial" (SCIR) approach is not only traditional in national democracies, but in North American universities as well.

> The most extreme form of the [SCIR] process would be a situation in which all parties made their decisions in terms of their unregulated self-interests, and upon receiving the effects of these decisions, make new decisions to deal with the problems revealed by the new information. No external collective sanctions would be imposed to force decisions to conform to values beyond those held by the individual.[2]

Figure 6, "Competing Paradigms of Decision Processes," adapted from Lindblom's "The Science of 'Muddling Through',"[3] is useful in describing and contrasting the two conceptions.

---

[1] Frank A. Schmidtlein, The Selection of Decision Process Paradigms in Higher Education: Can We Make the Right Decision or Must We Make the Decision Right?, Paper P-42, Ford Foundation Program for Research in University Administration (Berkeley: University of California, 1973).

[2] Ibid., p. 33

[3] Charles E. Lindblom, "The Science of 'Muddling Through'", Public Administration Review, Vol. 19, No. 2 (Spring 1959), p. 81.

Figure 6

COMPETING PARADIGMS OF DECISION PROCESSES

RATIONAL/COMPREHENSIVE/ PRESCRIPTIVE	SUCCESSIVE COMPARISONS/INCREMENTAL REMEDIAL
1. Clarification of values and objectives distinct from and usually prerequisite to empirical analysis of alternative policies.	1. Selection of values, goals and empirical analysis of action needed are closely intertwined.
2. The test of a good policy is that it can be shown to be the most appropriate means to desired ends.	2. Typically the test of a good policy is that analysts agree on policy without agreeing that it is the most appropriate means to an objective.
3. Means-ends analysis is comprehensive—every important factor is taken into account.	3. Analysis is limited. Important possible outcomes are neglected.
4. Theory is often heavily relied upon.	4. A succession of comparisons greatly reduces reliance on theory.

It is interesting to compare the process orientations of the two paradigms with respect to certain factors which are relevant to higher education decision processes. In the RCP paradigm, *change* is seen as intensifying the problem of prediction and thus creating a need to plan. *Deadlines* require the a priori analysis of events in order to identify and deal with critical variables. *Causal relations* are known or are discoverable through analysis. Goals and measurable *outputs* are essential to assess the accomplishment of objectives. Preciseness and manipulation of data requires *quantifications* of variables in order to develop effective models of reality. Critical decisions are made by central planners and policy makers, so central planning offices have high priority for *allocation of human talent and information systems*. The specification of goals, measurement of goal achievement, and analysis of causal relationships will lead to understanding and *consensus*.

By comparison, the SCIR approach considers *change* a factor which makes analysis complex and unreliable, and since plans are rapidly outdated, decision making must be remedial. Decision makers resist committing themselves in advance of *deadlines* in order to keep their options open and maintain flexible bargaining positions. *Causal relations* need not be known because effects are discovered through response to decisions. Means and *outputs* are determined simultaneously through bargaining. An unsophisticated emphasis on *quantification* can bias analysis by too great a concentration of attention on variables more easily quantified. Critical decisions are made throughout the organization so *talent and information* are required wherever decisions are made. *Conflict* is fundamental and inevitable, and goal clarification only exacerbates this conflict so "due process" and "rules of reciprocity" are means of conflict management.[1] With respect to value orientations, Schmidtlein has compiled an interesting list which is sampled by Figure 7, "Value Orientations."

Figure 7

VALUE ORIENTATIONS

VALUE DILEMMAS	RATIONAL/COMPREHENSIVE/ PRESCRIPTIVE	SUCCESSIVE COMPARISONS/ INCREMENTAL/REMEDIAL
Analysis/ Bargaining	Crucial facts are discovered through analysis.	Crucial facts are discovered through bargaining.
Clarity/ Ambiguity	Clarity improves the quality of decision-making.	Ambiguity aids consensus and maintains bargaining positions.
Expertise/ Legitimacy	Expertise is the critical requirement for effective decision-making.	Legitimacy is the critical requirement for effective decision-making.
Control/ Freedom	Consistent policies require the exercise of management control.	Bargaining permits individual discretion and avoids incapacitating conflict.
Accountability/ Autonomy	Responsiveness to the collective interest requires accountability to the central policy-makers.	Responsiveness to the collective interest requires freedom to bargain with a multitude of conflicting interests.
Collective/ Individual	Serving the collective interest is paramount.	Reconciling individual interests is paramount.

---

[1] Paraphrased from Frank A. Schmidtlein, op.cit.

Based on the ideas discussed in this section, it would appear that although each paradigm has its merits and perhaps its ideal sphere and field of application, neither, in its extreme form, can be considered the synthesis we seek. According to Hegel, "Every Concept is rational, is abstractly opposed to another, and is united in comprehension together with its opposites. *This is the definition of dialectic.*"[1] Franco Lombardi, in "After Hegel," observes: "Traditionally since Hegel, dialectic has meant the instituting of a discourse, generally of an historical or psychological-political-sociological nature, characterized by opposing theses dialectically to provide a synthesis."[2] Lombardi calls such a synthesis a "unique truth." Perhaps application of the Hegelian dialectic method of reason could reconcile the apparent contradictions into a more complete synthesis.

---

[1] Hegel Encyclopedia of Philosophy. Translated and annotated by Gustav Emil Mueller (New York: Philosophical Library, 1959) p. 82.

[2] F. Lombardi, "After Hegel" in New Studies in Hegel's Philosophy ed. Warren E. Steinkraus (New York: Holt, Rinehart and Winston, Inc., 1971) p. 245.

A CENTRALIZED STUDENT DISTRIBUTION SYSTEM

Gary Cooley
Assistant Vice-President
Analyticial Studies
University of Massachusetts, Boston

A CENTRALIZED STUDENT DISTRIBUTION SYSTEM

The major objective of a student distribution model is to further capitalize on the economies and efficiencies of centralized data systems by including the computer-based student distribution algorithm which attempts to optomize the fit between the student hierarchy of college preference and the admissions criteria and enrollments established by colleges. A distribution algorithm was proposed which has been developed on the basis of meritocratic access but at the same time realizes that meritocratic accomplishment is multi-dimensional.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

STATEWIDE MANAGEMENT INFORMATION

Michael Abbott
Coordinator
Planning and Research
South Carolina Coordinating Commission

STATEWIDE MANAGEMENT INFORMATION

Mr. Abbott's presentation dealt with the state-of-the-art in statewide Management Information Systems for colleges and universities, particularly that data which state agencies are collecting and utilizing, and future implications for this data in statewide planning.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

STRUCTURED PROGRAMMING FROM A MANAGEMENT PROSPECTIVE

Robert L. Mason
Director
Computer Center
Western Iowa Community College

STRUCTURED PROGRAMMING FROM A MANAGEMENT PROSPECTIVE

Structured programming! It's easy to pronounce. It's easy to spell. Why is it so difficult to define or at least obtain a good sense of direction for its use? This is the general feeling that somehow prevails now that structured programming has become one of those "buzz words" that are heard when programming discussions become technical. The concepts are discussed and there is usually some level of agreement as to its merit. Yet there seems to be little effort to employ the principles.

Before examining the specific principles of Structured Programming, perhaps we should examine the traditional professional training given our programmers. Programming as a professional occupation has experienced tremendous growth, but as yet, remains a craft and not a science. Just fifteen years ago, the programming field was still in its infancy stage.. Now, less than one generation later, there are thousands of programmers with varying degrees of education, training and experience. The field is becoming populated. But with what kind of programmers? Have you ever noticed how the majority of programmers become stagnant in the sense that they do not keep up with new developments? The intent here is not to put a bad rap on programmers. Indeed, all technical professions including doctors and dentists have a tendency to become outmoded technically. Programmers find a way that works and resist change. Why do you think that PL/I has never become popular? How come ALGOL has not received more support? Why do programmers continually try to make the situation fit the language? Imagine using FORTRAN to perform a list processing type problem. These situations are common and the inappropriate solutions are becoming the rule rather than the exception.

Where does that leave us today? Perhaps the blame connot and should not be placed upon the programmer. Typically they perform in an environment which is nonconducive to learning and implementing new concepts. And who has the responsibility for promoting the environment? Management does and this includes everyone from the senior programmers to the corporation presidents. What are their policies toward the advancing technologies? Are adequate training programs, seminars, and incentives provided? Possibly, a programmer receiving extra training each year might produce more in less time than the typical programmer could produce given the entire year. Additionally, this traditional method has a snowballing effect over the period of several years in that the entire programming environment becomes stale. The only influx of ideas and concepts is the result of employee turnover and this is not a beneficial situation. Management must become more aware and flexible in dealing with programming standards. There must be a genuine management effort directed toward upgrading present technology rather than merely hiring additional bodies. The point does not pertain to spending X number of additional dollars. but rather, providing a completely new and different emphasis to the present spending methods.

What is Structured Programming?

Perhaps one of the areas of technological education should be directed toward structured programming. Indeed, the concept has been praised as the greatest programming idea since the advent of the subroutine. Yet, only a small fraction of current programming population really understand and attempt to implement the concepts. Most programmers think that structured programming is merely GO TO free coding. Others feel that it is the development of code using the TOP DOWN approach. But it is more than this. As mentioned earlier, a definition is hard to derive but the fundamental message is to make the code as structured as possible by simplifing the control paths.

Program complexity is usually directly proportional to the number of jumps to other parts of the program. These jumps can be forward or backward or the result of mulitple entry and exit points within a routine. It becomes hard to visualize the present conditions at any given point in the code. What has or hasn't been executed is hard to determine. This leads to a state of confusion during maintenance. Code is modified; jumps are inserted; functions are deleted inadvertently. The code becomes a conglomerate of pieces and subsequently shop-worn. Further modification is like bailing water from a sinking ship. It's hard to tell whether it's going to go up or down.

Structured programming has as its creed; any program function can be performed using one of three control structures: 1. simple sequence, 2. selection, 3. repetition (see figure 1). Any kind of problem, decision, or logic can be accommodated using one of these three structures or some combination of the structures.

The concepts can actually be implemented by any programmer in any programming language. Granted the difficulty varies; but as a general rule, the higher the level of language, the easier the implementation. Assembly language coding becomes totally inadequate. The selection process needs the support of a IF-THEN-ELSE construct and the repetition process is governed by some form of the DO WHILE. While each routine represents some combination of these structures, it must be pointed out that each routine can have only one entry point and one exit point. This constitutes a further tightening of control and makes for more simplicity.

While these concepts are presented to arouse some awareness within the reader, the intent of this paper is not to explore in detail the structured programming concepts. However, there is a need to understand what has been attempted and accomplished in the field. This information is necessary for dynamic management decision making.

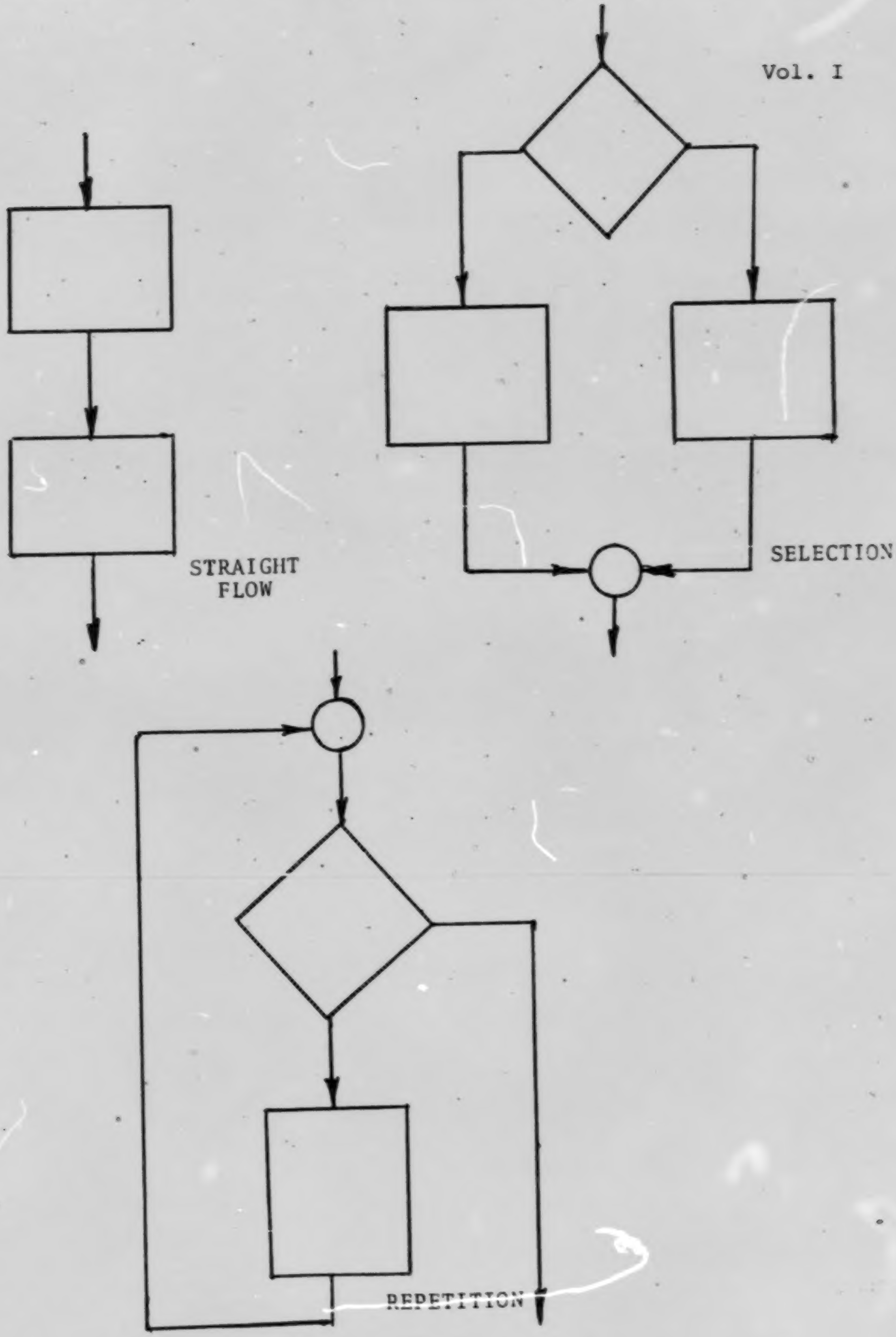


FIGURE I

The Direction of Structured Programming

The concept was "born" in the middle sixties. Initial credit is given to Böhm, Jacopini, and Dijkstra for the discovery. These people and other interested individuals were interested in the theorical aspects of programming. Proof of program correctness was their forte and structured programming became one of their primary tools. The papers written by these people have stirred great controversy; perhaps the greatest in programming history. It is now eight years later and the controversy continues. At the 1973 IEEE reliability conference, the division amoung programming advocates as to the merits of structured programming was split right down the middle. The resulting arguments were very interesting and very heated. Results of the largest single application of structured programming were presented by Dr. Harlen Mills of IBM Corporation.

In 1969, Dr. Mills and Mr. F.T. Baker established what they called the "chief programmer team" approach to system development. They experienced a great deal of success on the New York Times project (1969-1971). A complex information retrieval system was implemented using only a few highly skilled programmers under the direction of a chief programmer. Results were astonishing. In 22 months, the team "specified and designed the system and developed over 83,000 lines of original high-level language source code. By today's productivity standards for systems of comparable complexity, such a task would require several times the 11 man-years of effort actually used by the team." (1) Also, the system was put into production and performed remarkably. "In the first 13 months of operation of the on-line retrieval system, only one program error was detected that resulted in system failure." (1) The final results showed only one error per man-year of effort. Dr. Mills and Mr. Baker feel that "achievable targets for application programming are 10,000 lines of source code and one error per man-year." Both of these individuals attribute much of the success of the project to the structured programming concepts employed in the design and coding.

The results of these projects, coupled with recent shifts in emphasis regarding software development have led to modified goals. In the past, successful execution was the primary requirement. Was the output correct and timely? A second area of concern was software efficiency. Certain time and core size restrictions had to be maintained. Programmers tended to follow the rule "anything goes." There becomes a tendency for programmers to revert back to lower level languages to fulfill the core and time obligations. While it cannot be argued that a programmer can generate better machine code than the most sophisicated compiler, this does not yield long-range benefits for the programming department. Maintenance costs skyrocket to support this type of environment.

Management is now starting to recognize this situation. The emphasis is in the process of shifting to other factors, namely, reliability, maintainability and extensibility." (4) Of these factors, only reliability has been of previous concern and that was not to the extent necessary. That is why the numerous "system crashes." Other subsets of reliability namely proof of program correctness and error control have led to the development of structured programming as known today. The development has come from educational institutions engaged in research. The support came from those organizations dealing with high reliability systems such as in the aerospace industry.

These large scale projects have been very costly in both development and maintenance. Hence the motivation for new techniques to reduce cost. Structured programming appeared to have merit. First the theorists discovered the concept. Next the computer scientists sanctioned the procedure. Now it is time for implementation of structured programming at the administrative production level where maintenance and extensibility take on an equal importance to reliability. Why? It is understood that 35 to 50 percent of software costs can be attributed to maintenance. This is the single major area of software which can benefit from new techniques.

Program complexity causes problems during development, but picture the confusion during the maintenance period. The system must be modified due to the presence of a bug or a change in requirements. But alas, there has been a complete change over in personnel. How well does the new personnel understand the system? This program clarity or "understandability" is the ease with which a person can read code and determine the change to be made. Reduced complexity improves the understanding of code. Recently, a situation was encountered that bears out the problem of program complexity. An error was discovered in a 100 line FORTRAN program. This program was written by a top quality programmer who strived to keep the number of lines of code to a minimum. Therefore, the program became very complex with no internal documentation. Needless to say, when this same individual was asked to make a small modification, he needed as much modification time as he had used to create the program in the first place.

Complicated coding principles, redundant code, obsolete comments need to become a thing of the past. Structured programming is a manner of organizing, structuring, and coding complex problems. This in turn leads to easier maintenance; thus, reduced costs. Management must take the lead to inspire, motivate, or command its staff to learn and apply new techniques which will, in the long run, reduce the costly software maintenance and expandibility functions. When this happens, increased reliability will also emerge but is harder if not impossible to measure or observe in monetary terms.

What has been done and what can be done

The basic concepts of structured programming can be applied to any language. Differences become the varying degrees of difficulty. Of those procedure-oriented languages in wide spread use today, only PL/I and ALGOL are close to suitable for efficient use in structured programming. ALGOL has practically no acceptance in the United States leaving PL/I as the forerunner. This has prompted Daniel McCracken to predict "that within the three-to-five-year future, there will be, at long last, a swing to PL/I, precisely because it is well suited for structured programming." (7)

Different approaches to structured programming are also under current consideration. Many commercial software firms are designing their own structured languages. But this approach is shortsighted. What they really developed is merely a syntax. There exists no compiler for their language. A preprocessor is written which transforms a pseudo structured language into one of the current high-level languages. This means that a program has to undergo an additional transformation thereby resulting in even less efficient execution code. There is no support from the major vendors and who is to provide the necessary maintenance to their products. I don't envision any one software firm with the capability to provide this service. Therefore, it must be concluded that this is not the most beneficial approach.

The next logical step becomes the modification of an existing language and compiler. For example, let's add IF-THEN-ELSE and DO WHILE terminology to FORTRAN and eliminate all forms of the GO TO statement. Science Applications Inc. modified the IBM

FORTRAN G compiler to handle a structured language. The appendix to this paper illustrates this approach. Shown is a sort routine as it appears in the March 1969 issue of COMMUNICATIONS OF THE ACM. Both the original version and a modified SPL version were compiled and run several times an an IBM 370/145. The results are as follows:

	REGULAR FORTRAN SORT	STRUCTURED SORT
NUMBER OF RANDOM NUMBERS	5000	5000
CORE USED (BYTES)	1420	1542
TIME USED (AVERAGE OF 10 TRIALS)	3.435 sec.	3.46 sec.

Speed differed by less than one percent and core size by eight percent. There differences are small and worthy of notice and further exploration. However, the intent is not one of promoting this method but rather to stimulate thought as to the different approachs. This SPL language still has many shortcomings. Speed is slower; core size is less efficient; syntax is a bit ackward; and most noteable, it to has very little support.

Perhaps the best approach for management is to develop an entirely new language which would be comprehensive, yet simple to program, maintain, and extend. This method could probably only emerge if and when enough management pressure is brought to bear upon the computer manufacturers. Therefore, in all practicality, this also has to be ruled out for the near future.

What can be done?

Management has the obligation to be perceptive regarding programming standards. It is pertinent that a full understanding be obtained of the methods used within the shop. Perhaps a complete new approach is warranted. It is essential that management step back and give the situation a full and fair evaluation. It is easy to condone unsuitable standards especially when they were established by present management.

New techniques including the structured programming concepts deserve a chance. New training is probably necessary. Seminars can be offered. Money will have to be spent for education. If you provide one month of training yearly to each employee, that consists of an eight percent reduction in actual work force. However, if the new concepts can reduce the maintenance costs by fifty percent as some experts have predicted, this would have a net reduction of up to twenty-five percent of total software cost. IBM has experienced this situation. They are now setting new standards and providing additonal training to their people. It is going to be interesting in the next few years to view the results of those progressive managements who partake in the programming revolution.

REFERENCES

1. Baker, F. T., and Mills, H. D., "Chief Programmer Teams," *Datamation*, pp 58 - 61, Dec. 1973.

2. Bohm, D., and Jacopini, C., "Flow Diagrams, Turing Machines and Languages with Only Two Formation Rules," *Communication of the ACM*, pp 366 - 371, May 1966.

3. Dijkstra, E. W., "Notes on Structured Programming," April 1970, Academic Press, 1972.

4. Donaldson, James R., "Structured Programming," *Datamation*, pp 52 - 54, Dec. 1973.

5. Liskov, B.H., "Guidelines for the Design and Implementation of Reliable Software Systems," The Mitre Corporation, Report No. MTR-2345, Bedford, Mass., February 1973.

6. McCarthy, J., "A Basis for a Mathematical Theory of Computation," *Computer Programming and Formal Systems*, North-Holland Publishing Co., pp 33 - 70, 1963.

7. McCracken, Daniel D., "Structured Programming, An Overview," *Datamation*, pp 50 - 52, Dec. 1973.

8. Miller, Edward and Lindamood, George, "Structured Programming: Tap-down Approach," *Datamation*, pp 55 - 57, Dec. 1973.

9. Mills, Harlan D., "Mathematical Foundations for Structured Programming," IBM, FSC72-6012, February 1972.

10. Rader, R. J., and Douglass, C. C., "A Structured Programming Language Based on FORTRAN (SPLF)", Unpublished paper, Science Applications Inc., Huntsville, Alabama, June 1974.

PERSONNEL CONSIDERATIONS IN PROJECT MANAGEMENT

Charles G. Foskey
Associate Director
Administrative Data Processing
Uniersity of North Carolina at Chapel Hill

PERSONNEL CONSIDERATIONS IN PROJECT MANAGEMENT

Charles G. Foskey, Associate Director, Administrative Data Processing
University of North Carolina at Chapel Hill
Chapel Hill, North Carolina 27514

The purpose of this paper is to identify and discuss some of the people-oriented considerations that should be taken into account by managers of data processing projects. The paper addresses three broad categories of personnel considerations including pay and job perquisites, management theory, and specific management duties related to people assigned to data processing projects.

I. INTRODUCTION

The new five-track format for the CAUSE Annual Meeting is certainly a boon to attendees, but it makes tough on the speakers. We speakers are in the same position as the TV network executives in competing for prime time. Right now you have a choice of speakers from the University of Iowa, EIS in New Jersey, the University of Texas, or me. This situation is like going against Marcus Welby and Sesame Street. On the other hand, on my own network, I was fortunate enough to get slotted after the hot topic of structured programming and before one of the pioneers of University-oriented information systems, Dr. Weathersby, which is akin to having prime time between the Super Bowl and the TV showing of the Godfather. In any case, I thank you for coming; I feel that I will be able to make it worth your while.

As you can see, my topic is involved with Personnel Considerations in Project Management. I am not going to talk about mechanized systems for project management. There are a number of user developed systems and systems that are marketed by software houses that will assist you in the bookkeeping related to project management. Many of these mechanized systems work well, but none of them will actually do the job of managing. I am going to focus on some of the things that a manager must consider about people when they are plugged into a mechanized system.

I have divided the topic of Personnel Considerations in Project Management into three areas. The first of these areas is pay and perquisites; this category includes such things as salary, insurance, vacations, etc. I hope to be able to bring out both the importance and limitations of these items as they affect motivation and performance. The second area is involved with basic management theory or philosophy - here I am going to get at why we act as we do when we manage people. In part three, I am going to review with you some of the problems we have all encountered when it comes down to the nitty gritty of getting a group of highly intelligent people all pulling together toward the common goal of successfully implementing a computer-based data processing system.

2. PAY AND JOB PERQUISITES

The items of salary, insurance, vacations and holidays, retirement, company benefits and working space are all extremely important personnel considerations as they relate to job satisfaction and employee retention - but these factors are very limited when it comes to motivating people to get a job done. I am stating that these items are very important on the one hand and yet not so important after all. Let me expand on this by using the example of salary.

Douglas McGregor in his book the Human Side of Enterprise [1] writes that there is the "consideration of equity: whether the amount of money provided is perceived to be fair relative to the market, economic conditions, the importance of the job, and the individual's contribution." The equity consideration is very important. If the individual feels that he is being treated unfairly on pay day, this negative attitude will either cause him to leave the job or to work less effectively than he would otherwise. He must sense that he has equity at several levels.

For example, he must have equity with the industry. If programmers and analysts are earning significantly less than the prevailing wage rate, you have an unhealthy situation.

The individual must have equity relative to other organizations within the same company. If a campus has a professional group of programmers and analysts who make a contribution to the goals of the institution that is equal to or greater than accountants or other professional staff, yet they are paid less; the employees will have a reason to be unhappy.

Finally, the individual must have equity relative to other people in his own department. Even though two analyst/programmers doing equivalent work are well paid relative to programmer/analysts in other organizations and relative to employees in other departments, yet one earns considerably more than the other, the lesser paid employee is likely to develop a poor attitude which will reduce his effectiveness. It is equally unhealthy to have two employees with two different levels of responsibility who earn the same salary. Equity does not mean pay equalization; management must reward those with greater responsibility and productiveness.

Once equity is achieved, however, additional increments of pay will have only minimal effect on motivation. For example, suppose a project team that was already adequately paid was told that if they met certain deadlines during the coming year that, in addition to the expected cost-of-living and merit increases, they would also receive a bonus - then productivity may increase temporarily. The door would also be open to cheating on what constitutes "making" the deadline and create severe personal problems for those deemed to be not pulling their weight. If the individual is already adequately paid, money alone will not provide additional motivation.

A similar situation exists with other job benefits such as insurance. If you have inadequate group insurance and the spouse of a lead programmer gets sick, and as a result, the lead programmer is saddled with large medical bills - management has a problem. If management is perceived as being stingy with insurance benefits, this attitude can have a devastating affect on the project. On the other hand, adding to an already adequate insurance program will not necessarily increase productivity.

Similar statements can be made of retirement programs and vacation policy. The employee must feel that he is treated fairly. But once equity is achieved, additional retirement benefits or additional vacation days will not greatly increase employee motivation.

Working space is somewhat different from the other items that we have been discussing. The satisfaction and excitement of working with computer projects can often overcome the negative effects of inadequate salary and fringe benefits. Nothing can overcome poor working conditions, however. If you put your programmers in a noisy, crowded, poorly lit situation, you have to expect a loss in efficiency. The nature of systems and programming demands adequate desks, work tables, and book cases; to provide less is to assure inefficiency. Another aspect of space assignment is that of accessibility. The programmers must have a degree of privacy for those periods when they are thinking or trying to code. On the other hand, there must be a natural way to congregate to talk about mutual problems. Data processing professionals will steer bull sessions into problem solving sessions or learning sessions. In planning work space, management needs to consider that there are times when the programmer/analysts needs isolation and there are other times when it is important and even productive to shoot the bull.

Until now I have been discussing basic space needs that are a prerequisite to having an effective systems and programming staff. Niceties such as carpets, paintings, etc. are similar to other benefits - you need to have as much as your competition but beyond the level considered to be equitable, additional plushness of carpet, square footage, etc., will yield little gains and perhaps even decrease productivity.

I would like to summarize the section on pay and perquisites by stating that this is an area where one must be competitive. If not, low morale, low productivity, and high employee turnover are the inevitable results. But if salary and benefits are increased beyond the point of fairness to your employees, employee turnover may be reduced, but there will be little increase in productivity. Motivation of employees requires more than good pay, and this leads us into the next area of discussion.

3. MANAGEMENT PHILOSOPHY/THEORY

Many individuals in government and industry are turned off by the words, "management theory," and with good reason. All of our working lives we have been subjected to buzz words describing schemes that will aid management in getting the work out of the shop. Theories have been poorly thought out, or poorly implemented, or both. The practical manager may say that he has little use for theory. But our friend Douglas McGregor [1] states that you cannot escape theory - you'v got it whether you want it or not. McGregor reasons that every act a manager makes is based on some assumptions that one has about human behavior i.e. if I do a then b will occur.

McGregor [1] has popularized two terms about management theory and these are simply theory X and theory Y. Theory is not to be confused with specific tools such as management by objectives, decentralization, democratic leadership, consultative supervision etc. All of these tools can operate under either theory X or theory Y. The thing that characterizes a management style as being either theory X or theory Y are the assumptions that management makes about human behavior. Theory X is identified by three assumptions:

"1. The average human being has an inherent dislike of work and will avoid it if he can.

2. Because of this dislike of work, most people must be coerced, controlled, directed, and threatened with punishment to get them to put forth adequate effort toward the achievement of organizational objectives.

3. The average human being prefers to be directed, wishes to avoid responsibility, has relatively little ambition and wants security above all."

Let's contrast these assumptions with theory Y.

"1. The expenditure of physical and mental effort in work is as natural as play or rest. The average human being does not inherently dislike work. Depending upon controllable conditions, work may be a source of satisfaction (and will voluntarily performed) or a source of punishment and will be avoided if possible.

2. External control and the threat of punishment are not the only means for bringing about effort toward organizational objectives. Man will exercise self-direction and self-control in the service of objectives to which he is committed.

3. Commitment to objectives is a function of the rewards associated with their achievement. The most significant of such regards e.g. the satisfaction of ego and self actualization needs, can be direct products of effort directed toward oganizational objectives.

4. The average human being learns, under proper conditions, not only to accept, but to seek responsibility. Avoidance of responsibility, lack of ambition, and emphasis on security are generally consequences of experience - not inherent human characteristics.

5. The capacity to exercise a relatively high degree of imagination, ingenuity, and creativity in the solution of organizational problems is widely, not narrowly, distributed in the population.

6. Under the conditions of modern industrial life, the intellectual potentialities of the average human being are only partially utilized."

These six assumption of human behavior are radically different from the three assumptions of theory X. Suppose we say well, all right, theory Y looks pretty good. What does it do for me when I manage programmers and analysts. First of all, lets talk about what it does not mean. Theory Y does not imply permissiveness. Many individuals have all of the theory X assumptions about human behavior yet adopt permissive attitudes towards their subordinates as a strategy to manipulate them. Theory Y does not mean abrogation of either authority or responsibility. Theory Y managers have the same authority to hire, fire, plan, budget, execute, and evaluate as theory X managers. The responsibility for the success or failure of data processing projects lie with the theory Y project leader as well as the theory X project leader. What is the difference?

Theory X managers use the authority they have to directly control the activities of their subordinates in order that the organization meet its objectives. Theory Y managers use their authority to create conditions that will permit goals of the organization to be achieved as a direct result of each individual in the organization achieving his owns goals. Simple - right? - wrong; theory Y is easy to talk about, yet hard to accomplish. In the first place it is very difficult to know yourself. You may think you have theory Y assumptions, yet the people who work for you may believe you are a strong adherent to theory X. Secondly, you may know you are theory X, you may also know that you have achieved success, and you have no motivation to change. Finally, you may sincerely want to incorporate theory Y thinking into your own management style, yet you may simply not know how to go about it. You may, for example, confuse permissiveness or abdication of responsibility with theory Y.

McGregor suggests a framework that could lead a manager into theory Y.

1. Communicate to your employees concisely what the general objectives of the organization are - i.e. to provide more information, to cut costs, to provide accuracy etc.

2. Ask each subordinate to examine his own goals in light of the overall goals of the organization.

3. Act as a sounding board during the period when the employee is implementing his own objectives.

4. Ask the employee to evaluate the results in terms of his original objectives. Ask the employee to suggest specific items which would have improved performance.

The problem of converting to theory Y is still a formidable task; My boss does not always act under the assumptions of theory Y, nor do I, nor do my subordinates. One of the reasons for this is that the theory itself has some weaknesses.

Theory Y talks in terms of the "average" person: i.e. If you have a management strategy compatible with theory Y, then the average employee will respond in a positive manner. The next question is, "what if none of my employees are average?" Good question. I am not attacking theory Y; I am merely trying to get at one of its major weaknesses; and that weakness is the idea that, although here are individual differences in the degree of self-motivation, most differences are the result of the managerial environment - change the environment and the individual will change. This latter concept is not always valid, and this leads me to my next point which is: A good management strategy accounts for individual differences. Since the beginning of organized research in psychology and even sociology, there has always been a great deal of emphasis on individual differences. Yet both those who develop and those who practice management theory frequently do not realize the importance of individual differences. Gerry Bell [2] a professor at the UNC-CH Business School has written a newly published book in which he attacks this problem. Bell did a thorough study on some 3,000 business and government leaders and claims that he has been able to identify some six types of personalities; and that almost everyone, including managers, can be classified as falling into one of those six categories.

The categories are as follow:

1. The Avoider
2. The Attacker
3. The Pleaser
4. The Commander
5. The Performer
6. The Achiever

Lets take a few minutes here and look at the characteristics of these groups of people.

"1. The Avoider

a. The avoider needs to avoid failure.
b. His goal in life is to hide.
c. He wants to stay out of trouble.
d. He is dependent on others to lead him.
e. He possesses little self confidence.
f. He prefers assignments that are stable and routine.
g. He takes few personal risks and sets low goals for himself.

2. The Attacker

a. The attacker needs to release his hostilities without accepting responsibility or dependence on others.
b. He likes to hit others and to run from obligations.
c. He tears down a plan but offers no solutions.
d. Being an authority rebel, he loves to argue and to debate.
e. He is sarcastic, cynical, negative.

3. The Pleaser
   a. The pleaser needs to make others like and approve of him.
   b. He seeks acceptance from all with whom he associates by being kind and generous and by going along with others.
   c. He would rather make friends than perform a task well.
   d. He prefers assignments that involve dealing with people on an easy sociable basis.
4. The Commander
   a. The commander needs to control whatever situation he confronts, dominate every group he is in, and live an orderly systematic life.
   b. He dislikes ambiguity and uncertainty.
   c. He sees his world in clear cut categories.
   d. He approaches new situations in a dogmatic and stubborn manner.
5. The Performer
   a. The performer needs to gain prestige and recognition.
   b. To do this he maneuvers and tries to make a good impression.
   c. He strives diligently to be proper and respectable.
   d. He changes his values to go along with the most advantageous positions.
   e. He is very hard to pin down.
   f. He seeks those tasks which maximize his image and prestige, regardless of whether he believes in the tasks.
6. The Achiever
   a. The achiever needs to maximize his potential to reach his highest levels of competence to become self-fulfilled.
   b. He is problem centered and goal oriented rather than self centered
   c. He has much self confidence and self reliance.
   d. He works toward challenging yet realistic goals.
   e. He accepts feedback about his results and likes to explore creative new ways to reach his objectives.
   f. He devotes himself to things that he believes in, not those that merely give him prestige and money."

I have mixed emotions about the work of Bell. On the one hand, I feel that it is dangerous to try to categorize a person (especially one you do not know well) and then to expect him to react in accordance with the traits of the category in which you have mentally filed him. But what is so compelling about Bell's book is the uncanny way in which the categories describe the traits of you, your co-workers, and I. Bell does admit that many people are a mixed bag of personality traits. For example, a person may be 40% achiever, 40% commander and 20% performer. But Bell also says, in effect, that as you get to know someone, their dominant traits will emerge; and if you are an astute observer, you can then fit the individual into one of the six categories.

How does the work of Bell [2] relate to theory X and Y? Lets look again at the traits of the achiever.

"a. The achiever needs to maximize his potential to reach his highest levels of competence to become self-fulfilled.
b. He is problem centered and goal oriented rather than self centered.
c. He has much self confidence and self reliance.
d. He works toward challenging yet realistic goals.
e. He accepts feedback about his results and likes to explore creative new ways to reach his objectives.
f. He devotes himself to things that he believes in, not those that merely give him prestige and money."

It is apparent to me that these traits and these traits alone, would lead to assumptions compatible with theory Y. Yet of the people studied by Bell, only 8% of the general population, 14% of middle management, and 24% of top level management are achievers. Traits exhibited by the other five types: The avoiders, attackers, pleasers, commanders, and performers are all compatible with theory X. Yet these people - if Bell's study is representative - comprise 92% of the general population 86% of middle management, and 74% of top management. i.e. Theory Y will not work, because even if you try to create a situation in which the goals of the organization are achieved as a result of achieving the goals of the individuals - the goals of the avoiders, attackers, etc. which comprise the vast majority of the population will not coincide with the goals of the organization.

If I stop now and say nothing more, you may think that I recommend that you stick with whatever management strategy you have now have - there is simply no use in trying. I am not saying this at all. I am saying, however, that I cannot tell you exactly what to do in regards to management theory. I don't have your boss, your subordinates, your computer, or your problems, I don't have your education, your background, and your experience. But I am going to recommend that:

1. You do some reading on the subject.
2. You analyze the underlying management theory that exists in your University and your department.
3. You observe the behavior of your boss, your peers, and your subordinates, and try to understand why they act as they do.
4. You make a concious effort to evolve toward becoming a more effective manager and, hence, a more effective person.

4. THE FUNCTIONS OF DATA PROCESSING MANAGEMENT

Let's be done with theory and get down to some day-to-day personnel considerations about which we are all concerned. The first of these is staffing.

One of the toughest problems facing project leaders with hiring responsibility, is recruiting and evaluating applicants. Nothing can contribute to the sucess of a project more than the decisions one makes when hiring. There are many different aspects to evaluate in hiring but I want to talk about four which I consider to be important.

1. The personality traits of the prospect.
2. His raw aptitude and ability.
3. The kind of experience that he has had and
4. His record of success.

Gerald Weinberg, writing in the Psychology of Computer Programming [3], identifies six personality traits that are critical to programming and which I feel are important to systems analysis as well. These traits include the ability to tolerate stressful situations, adaptability to rapid change, a modicum of neatness, a measure of humility, assertiveness, and a sense of humor.

Anyone with experience in data processing will agree that these traits are important. For example:

1. The ability to tolerate stress: Anyone that has not had to tolerate stress for a prolonged period, has not really experienced data processing. Good management will prevent stress from becoming a way of life, but stress is a normal part of data processing, and it is to be expected.
2. The ability to change: If an individual is not adaptable to change, he will tend to design inflexible systems, and his technical capabilities will rapidly become obsolete. Rapid change is an always-present, controlling feature in our business, and those who have a low toleration for change will have a rough time in data processing.
3. The third trait discussed by Weinberg was that programmers and analysts must have a modicum of neatness. Sloppy work, equates to sloppy systems operations: enough said.
4. A trait frequently not recognized in our business is humility; yet humility is a must in systems design and programming. One must believe he can make errors, or there will be no serious effort made to find them.
5. On the other hand, programmers and analysts must be assertive. If they are not willing to challenge the boss, to challenge others, to overcome obstacles, then they will not be effective members of a project team.

6. Finally, Weinberg recognizes the need for a sense of humor. If one is doing anything of consequence with a computer, the computer will frequently force the programmer into making stupid mistakes. If project members are unable to laugh at themselves, they will likely have a bleak life in dealing with the machine.

Since we have been able to identify the necessary traits, it does not necessarily follow that it is easy to devise tests so that we can always hire those people whose personality fits the job. There have been experiements in developing personality profiles for programmer/analysts and in developing tests that indicate the degree to which individuals match the profile. These tests have had mixed results and are subject to misuse and abuse. I feel that the manager is better off simply knowing what the desired traits are and then training himself to recognize these traits.

The second aspect to consider in hiring someone is their analytical and programming ability. The problem here is different from that of identifying traits. Yet it is equally difficult to solve. One of the most widely used is the famous Programmers`Aptitude Test or PAT. There are some serious problems with the PAT. One problem is that it is so widely used that many people have taken it more than once. Experiments have shown that, contrary to the desired result, "aptitude" tends to increase each time a person takes the test or a variation of the test.

Furthermore, the PAT does not necessarily predict good performance. Some correlation studies have shown a relatively high correlation coefficient between PAT scores and performance. Others have shown only that a positive relationship exists between test scores and performance in programmer training. Yet many people who test poorly, and have only an average record in school, perform very well on the job. Perhaps the very best that the tests can do is eliminate individuals who absolutely cannot learn to program.

Many companies use the PAT or similar test to test, not only inexperienced programmers, but experienced programmers as well. Weinberg [3] states that he would rather test an experienced programmer by having him write a program.

Of course the problems of identifying aptitudes and skills of analysts is even more difficult than that of programming, and there has been less written about experimentation in this area. My only advice is to stay abreast of the literature and open to any proven methods of testing, and to be wary of those methods which are merely new.

The third aspect of hiring is looking at the depth and kind of experience that a person can bring to a job. Typically interviewers look for the following when talking to prospects:

- Experience with what machine? core size?
- Experience with what operating systems?
- Experience with what languages?
- Experience in what industry?
- Experience in what applications?
- Experience with what access methods?
- Experience with which data base management system?
- Experience with telecommunications
- Experience with computer-output-microfilm
- Experience with interactive program debugging

What managers must consider, of course, are ways to balance the project team. Weaknesses in one area by Analyst A must be balanced by strengths in the same area by Analyst B. The individual manager when hiring must decide which factors are important and which factors carry little weight.

The fourth and final aspect to be considered in hiring is quite simply the record of success the candidate brings to the job. The answer to this question as well as to the question of the kind and depth of experience lies in the old standbys: interviews and references. I recommend a thorough check on references. There are a large number of people, some of them in data processing, who are floating around with false credentials. Many companies simply do not bother to check.

In summary, I would like to reiterate that, in hiring, the alert manager will consider, personality traits, aptitude, breadth and depth of experience, and record of success. Each manager must work out his own methods of evaluating each of these criteria.

After staffing, training is obviously an important personnel consideration for project management. While a project is underway, however, the manager might assume that if he is working with experienced people, there is no need for training. This is usually not the case. Training needs should be recognized when a project schedule is being developed. For example, some of my people are currently implementing an on-line system for the University Cashier's Office. The system will use Burroughs programmable teller terminals and a programmable mini-computer as a communications interface. The terminals and mini-computer have a special COBOL compiler as well as an assembler-like language. Here is an obvious situation in which you have to allot time for training. Other training needs are not so obvious.

Most manufacturers offer a wide variety of service routines, language compilers, debugging aids, specialized interactive languages, retrieval systems, and data base management systems. Many programmers and analysts have a thorough knowledge of some of these areas and only a cursory understanding of others. Vendor software is by no means a trivial training problem for the project manager.

Some project members may lack basic conceptual training in broad areas such as telecommunications, data base management, or structured programming.

The project manager needs to evaluate the training needs of each member of the project team. He must evaluate those needs in terms of the goals of the project. If gaps in training will directly affect the success of the project, then the project schedule must be expanded to include the training. If the gaps in training are in areas of only peripheral concern to the project, the manager may choose to defer training in these areas.

Once these needs are identified - how do you fulfull them? The one tried and true method is OJT or on-the-job training. OJT is the most widely used method in the industry. Even in large companies with formal training for their data processing staff, OJT is put to use daily - this as it should be. However, OJT has its limitations. OJT is spotty: people tend to learn only the subjects they want to learn. OJT is shallow: people tend to learn only to the depth that they consider relevant. OJT is infectious: bad programming habits can spread as easily as chicken pox in the second grade of a grammer school.

There are several alternatives to supplement OJT. There are many good programmed instruction manuals available.

Several firms have produced video assisted instruction (VAI) courses on a broad range of data processing subjects. VAI consists of video and audio tapes combined with selective readings. VAI is not a cure-all for training, but it can be useful.

Formal in-house courses and seminars are another useful method.

Vendor courses, and seminars similar to the ones that you are currently attending all have a place.

A manager with imagination will analyze the needs of each member of the project team and then use the resources available to him to meet that need. It is not enough to say that an organization has neither the time nor the money for training - if this is so, then the organization probably does not have the resources to be in data processing.

Lets suppose for a moment that we solved some of the problems we have discussed. We have worked on the problems of equitable pay and benefits, we have done our hiring, we have made plans for an on-going training program. We have been immersed in the concepts of theory Y, and we have become sensitive to individual differences. Having done all this, how does a manager make the project go? I am not going to discuss the technical framework associated with project management. The forms, the check points, the continuing feedback that comprise mechanized project management systems are all important and cannot be ignored, but my topic is personnel coniderations in project management. And the overriding personnel consideration is that of getting the goals of each of the individual team members aligned with the goals of the project. Let me repeat: the overriding personnel consideration in project management is that of aligning the goals of each project member with the goals of the project.

This need for goal congruence may seem obvious to many people, yet somehow this aspect of management is frequently overlooked.

There are many aspects to consider in gaining goal congruence: The first is to allow the team members, to the maximum that is practical, input with regards to determining the overall objectives of the project. A person who helps to decide what objectives the proposed system should accomplish and also how the system is to accomplish it, will likely be highly motivated to see it through to a successful completion. Once the overall objectives and general design of the system have been completed, perhaps the system can be divided into logical components such as input processing, file updating, data retrieval and report preparation. Within each of these areas the programmer/analyst should again be given the maximum practical latitude for design and implementation. At another level, the major system components can be broken into programs or program modules. Even at this level, the programmer can be encouraged to develop imaginative solutions to problems facing him. The manager should continually strive to create a situation in which each member feels that he "owns" a part of the project.

You may be sitting there thinking that I am recommending automated anarchy. If you give people maximum latitude in developing system components, how do you know that the components will fit together and work as a unit? Here is where the project manager must come to the fore. By a judicious mix of communication, education, persuasion, and a modicum of authority, the goals of the designers of each of the program modules will become aligned with the goals of the system components which in turn become aligned with the objectives of the total system.

Let me also state that I am not recommending that we throw out standards for system development and programming. Good standards serve only to reduce bad practice. If, in developing system components or programs, an analyst or programmer feels the need to violate the standards - then management has a training problem in that the individual does not understand the reasons for the standard, or you have a bad standard which should be amended. Good standards direct the imagination of programmer/analysts away from practice that experience has shown to be non-productive and frees their minds to concentrate on better solutions to the problems at hand.

Allowing programmers and analysts a degree of control over systems and program development is one way of assuring that these people have some control over their own professional life. There are other ways.

One method that has worked for me is to have the individual identify gaps in his own data processing experience and then to look for project assignments that will help in filling in these gaps. For example, if there is a need for Computer-Out-Microfilm (COM) and there is an employee who wants experience in this area, the project manager may wish to make an effort to assign him the task of making COM work. The same can be true for virtually any area of data processing. If people are convinced that they are growing professionally and can point to specific areas where they can see growth, then congruence is assured. Common sense dictates limitations to this concept. One cannot assign a man with six months experience the task of designing the telecommunications component of a major system. But, if the manager uses his imagination, assignments can be found that will coincide with the career objectives of the individual.

If the programmers and analysts feel that successful implementation of the total system will satisfy their own needs for self esteem and recognition, and further that the project is aiding them in achieving their career objectives, then the manager has licked the problem of motivation - because the people will be largely self-motivated.

Another personnel consideration is that of project estimating - it may appear that this is strictly a technical problem. This is only partially true. It is true that we need better methods for project estimation, but it is equally true that we need to know why people estimate the way that they do.

In a recent magazine article, Dr. Fred Brooks [4] has published some excellent insights into the problems of project estimating. The essence of

Dr. Brooks ideas on the subject can be summed in a quote from the Menu at Antoines, one of the finer restaurants in New Orleans, and it reads as follows: "If you are made to wait, it is to serve you better and to please you." Dr. Brooks states that people in data processing are simply too nice, or more bluntly, they are gutless when it come to estimating. How many meetings have there been at which someone asks, "How long will it take?" "40 man-months," comes the replay. "That is ridiculous" shoots back the questioner. "It is inconceivable that this project can take that long." "Well all right, how much time can I have?" Obviously, this method of estimating leaves something to be desired.

Another variant is, "We have to be ready by the beginning of the fiscal year" to which comes the reply, "Gosh, I do not know whether we can or not; I will do my best." This kind of thinking results in a situation where papa bear projects, mama bear projects, and tiny little baby bear projects, all with different starting times, are all due to be completed on June 30th.

Why do we do this? When we are new at the business, we do it because we simply do not know any better. We do it because we feel that it is expected of us. Dr. Brooks feels that we also do it because we tend to be optimistic about our work, we make the assumption that "if all goes well" we should be able to complete task X by next Tuesday, or a week from Wednesday, or in time for a board meeting. Yet, we all know that everything is not going to go well. The programmer on the project may say he is 90% through testing - how does he know? Until the program can handle a set of test data at least deemed to be exhaustive, you do not know whether you are 5% complete or 95% complete. Uncovering bug X may uncover bug Y, correcting bug A may create bug B - we simply cannot assume that all is going to go well.

Another very human problem is that we too often confuse effort with progress. If we see people working late and on weekends we say to ourselves, "those guys are really getting with it; they may be through by Thanksgiving after all" - not so. I have been in situations where I have worked very hard, yet project completion seemed to be receding rather than getting closer.

Dr. Brooks [4] points to another severe problem that occurs when you are dealing with project estimation. And that is the confusion between people and man-months - the two are not the same. Yet when a project falls behind, the natural tendancy is to throw more people into the breach. Suppose you have three people working on a 12 man-month project scheduled for completion in four (4) months, can you say that six people can do the job in two months? Or that twelve people can do the job in one month? Or that 250 people can do the in one day? I think you can answer that - the problems of training and communication are so acute that, beyond a certain level, adding people will not only fail to help, but may actually serve to slow things down.

The converse of being too eager to please is something call "sandbagging". Sandbagging works like this: The customer walks in and says, "This report is excellent, but I am meeting with the deans and department heads in two weeks and I need the same report but excluding all pass-fail hours."

The data processor looks at the report with a worried frown and says, "Gee, I don't know. This is a heavy program change and because of our previous commitments, machine time is scarce. Let me have the job and I will do what I can, but there is no way that I can promise you anything within two weeks." That afternoon the programmer adds an "IF" statement, recompiles the program and produces the report in a single job stream. One week later he calls the user and says, "Well that was really tough, but I struggled and got it done one week early."

We have talked about some problems involved with project estimation.

1. Gutless estimating - we want too much to please
2. Over optimism
3. Confusing effort with progress
4. Mythical man months
5. Sandbagging

All of these problems can be partially cured by a mechanized project management system. The rest of the cure relates to understanding why people act the way that they do, and then leading them out of their bad estimating habits.

1. The problem of estimating to meet the needs of the customer (gutless estimating) is overcome by making programmer/analysts understand that the time required to complete a given task is independent of the available time between now and the first of the fiscal year, or now and when the board meets, or now and when the new building is completed.

2. As indicated above, Dr. Brooks feels that our over-optismism is based on the assumption that all is going to go well. People assume that all is going to go well because, if their work contains unexpected flaws, they, themselves have unexpected flaws. A defect in system design may equate to a defect in the systems analyst. It is our job as managers to make people understand that flaws are to be expected and time should be allowed for their correction.

3. To the person who confuses effort with progress - tell him firmly to look not at the sweat on the furrowed brow but to look instead at milestone events that are 100% complete.

4. The person who confuses people with man months should be reminded that a 12 man month task cannot be completed by twelve people working one month. For each additional human man month you add, you are adding only X% toward productive work and 100 - X% toward communication with the other project members.

5. And finally, the sandbagger must be told again and again the story of the little boy who cried wolf.

The last of the personnel considerations related to the nitty gritty of managing projects involves communication. Everyone talks about communication, Gerald Weinberg talks about communication, Fred Brooks talks about communication, Douglas McGregor talks about communication, Gerald Bell talks about communication. I dare say that there is not a business management textbook in use today that does not deal at great length with the problem of communication.

All of the pop-psychology books that deal with inter-personal relationships such as I'm Ok You're OK, The Intimate Enemy, Open Marriage, Parent Effectiveness Training, and so forth have a recurring theme, and that is the need for better communication.

In listening to the Watergate hearings, one gets the clear understanding that as information moved from level to level, the content of the information become altered. A would tell B, B would tell C, D would tell E and so on, but Mr. A,B,C and D would not all talk together about the same subject. Lack of communication was one of the big factors that led the administration into Watergate.

But even while operating in a framework well-defined by a project management system, it is well to be aware that there are aspects of human nature that are mitigating against good communication. Documents tend to describe things as they should be rather than as they are, milestone events including programs with "only one outstanding bug" are listed as 100% complete. Problems brought up in meetings are smoothed over rather than solved. Meetings can lapse into unproductive sessions of finger pointing and defensiveness. Project status meetings can come to be typified by the motto: "If you fall behind, blame someone."

To avoid this kind of quagmire, management must establish a climate in which error is accepted and even expected. The individuals should be rewarded for facing up to honest mistakes and overcoming them. The manager should also realize that an analyst or a programmer has a lot of his ego tied up in a project and that there are tactful ways of honestly attacking a problem without attacking the individual.

5. SUMMARY

To summarize, it would be useful to identify a major obstacle faced by systems and programming project managers in keeping personnel considerations in perspective: We frequently use mechanized management systems to help us as we develop other mechanized systems. In such an environment, people considerations can get lost in the shuffle. Avoiding the trap of omitting people concerns takes a concious effort on the part of project managers.

Project managers must be aware of the strengths and limitations of pay and benefits in employee motivation. Management needs to examine its own assumptions about people, its philosophy in dealing with people, and whether the goals of the organization's employees are aligned with the goals of the organization. Finally, management needs to effectively apply good theory to the elements of staffing, training, motivation, project estimation, and communication which comprise the significant personnel considerations of project management.

REFERENCES

[1] Douglas McGregor, The Human Side of Enterprise, McGraw-Hill, New York, 1960.

[2] Gerald D. Bell, The Achievers: Six Styles of Personality and Leadership, Preston-Hill, Chapel Hill, 1973.

[3] Gerald M. Weinberg, The Psychology of Computer Programming, Van Nostrand Reinhold Co., New York, 1971.

[3] Frederic P. Brooks Jr., The Mythical Man-Month, Datamation, vol. 20, no. 12, December, 1974, 44-52.

INQUIRING SYSTEMS FOR HIGHER EDUCATION: A Perspective Based on Theories of Development of Organizations, Individuals and Decison Structures

George B. Weathersby
Associate Professor
Harvard Graduate School of Education

INQUIRING SYSTEMS FOR HIGHER EDUCATION: A Perspective Based on Theories of Development of Organizations, Individuals and Decision Structures*

by
George B. Weathersby
Associate Professor
Harvard Graduate School of Education
April 1975

Introduction

This paper speculates on some relationships among the patte organizational development, individual development, decision structu and information systems in higher education. In the decade since the Berkeley Free Speech Movement, colleges and universities have constantly been charged with ineffective or inefficient leadership. In a recent survey[1] of a large panel of leaders in higher education there was widespread agreement that institutional leaders do not initiate the most important changes in higher education, that they do not have meaningful information as a basis for decision and that they are viewed as the major resistors to changes which almost everyone (including fellow administrators) agree should occur.[2]

In their study Cohen and March (1974) described the college presidency as reactive, parochial, conventional, self important and illusory. They labeled colleges and universities as "organized anarchies" with problematic goals, unclear technology and fuzzy boundaries. In a similar analysis reported in a recent issue of Daedalus, Adam Yarmolinsky (1975) argued that colleges and universities have, remarkably, survived "without developing any conscious process for making institutional choices." (p. 61) Yarmolinsky went on to argue that the paralysis of decision making in higher education results from four major disfunctions between faculty and administration: substantive planning and budgetary decision making, departmental structures and functional areas of university concern, and the theory of direct democracy and its practice in the university (p. 61). Yarmolinsky concluded that four major preconditions for effective institutional decision making are: "a comprehensive management information system; a reasonable division of responsibilities among administration, faculty and students; an acceptance by faculty of representative government, as a working principle; and the legitimation of university administrations in the eyes of faculty--and, *pro tanto*, in the eyes of students" (p. 64).

This paper is about information (or, more generally, inquiring) systems for the management of higher education. Institutions of higher education are viewed as organizations composed of individuals, some of

* To appear in the CAUSE Conference Proceedings, 1975.

whom are in positions of responsibility and authority. These individuals perceive the world in ways consistent with their own personal development. The characteristics of the organization and the individuals jointly determine the most appropriate type of decision making and information systems. Currently available computer based information systems are of little real use to decision makers[3] and understanding the interrelationships among organizational and individual development, decision structures and information systems can provide a solid conceptual basis for the design of truly useful management information systems.

Colleges and Universities as Organizations

We have inherited an organizational structure in higher education that has evolved without too much conscious thought. Recently, there has been a growth of interest in the study of universities and colleges as organizations that are not terribly different from other organizations.[4] Based on his study of corporate organizations, Greiner (1972) identified five phases of organizational evolution in which each phase is characterized by the management style most conducive to growth of development; correspondingly, successive phases are reached by successfully resolving a dominant management problem inherent in each phase. Although Greiner's results are based on corporations, I believe the same general pattern is very suggestive of organizational evolution in higher education.

Greiner argues that organizations linearly evolve (an assertion I will subsequently question) and that each phase ultimately ends in a predictable management crisis the resolution of which requires the top management decision making style to evolve to the next phase. These phases occur over the life cycle of an organization at different rates depending in part upon the external environment: when resources are relatively plentiful decision making crises can be postponed and an evolutionary phase prolonged; conversely, when resources are increasingly scarce managerial crises are often precipitated which lead to more rapid evolution of decision making style. Furthermore, the phases are not normative; higher phases are not of greater value or prestige--they reflect the changing appropriateness of different top management decision making styles. Greiner's five phases and transitional crises are shown below:

Table 1: Organizational Phases and Transitional Crises

Phase of Dominant Management Style	Transition by Dominant Management Crisis
1. Creativity	Leadership
2. Direction	Autonomy
3. Delegation	Control
4. Coordination	Red Tape
5. Collaboration	?

Source: Greiner (1972), p. 41.

The life cycle of a higher educational organization could be described by these phases as follows. In phase 1 a group of creative individuals come together to start a new program, a new campus or a new institution. Usually a small number of very dedicated individuals share a dream of what the program, campus or institution could be. They pour enormous energy into getting the activity started. Decision making is often collegial. Communication is informal and unstructured. The organization can be very responsive to its immediate clientele. However, as the organization grows and more individuals become involved, informal communication and collegial decision making become increasingly ineffective and time consuming, external agencies expect a single spokesperson for the new program, and the first major decision to merge with another institution , to become accredited, to apply for or accept philanthropic support, to release a faculty member, to descipline a group of students, the retirement of the founding president and the ensuing search for a successor, the imminence of financial collapse, or whatever precipitates a crisis of leadership; who will be responsible for pulling the organization together and acting on behalf of all its members?

Those institutions that survive this first crisis usually do so by becoming more directed (phase 2). These phase 2 roles and responsibilities are more functionally organized (for example, academic, business, student affairs, or personnel officers), transactional accounting systems are installed, work load standards or expectations are adopted, communication becomes more formal as a chain of command is established. As this functional structure for implementation of decisions evolves, top management becomes increasingly remote from the classroom, laboratory, library or other student-faculty learning interaction and both lower-level managers and the creative people originally attracted to the organization feel stifled by centralized bureaucracy which often appears to be out of touch with academic reality. This precipitates the autonomy crisis: who should make the operating level decisions? Top management relaxes centralized control only reluctantly because previously they were successful autocrats. Meanwhile the most creative, mobile and dissatisfied faculty, students and administrators leave for less restrictive (phase 1) environments.

Institutions that survive the crisis of autonomy usually do so by increasing formal delegation (phase 3) of authority and responsibility. Programs, departments, colleges, campuses are all given more authority to deal with day-to-day management and to formulate their own goals. Top executives manage by exception and focus on new program development and overall resource acquisition from state legislatures, philanthropy, private individuals (alumnae and others) and establishing tuition and student aid (pricing) policies. However, as each subunit goes its own way, establishes its own standards, and develops its own identity independent of other subunits, the disparate subunits collectively precipitate the control crisis: how can top management maintain accountability and coordinate the actions of autonomous subgroups?

Similarly, institutions that survive the crisis of control usually do so by developing formal organizational and managerial systems of coordination (phase 4). In phase 4 top management usually expands the central staff to develop planning, information, control, evaluation, personnel and other "systems" that attempt to transfer the initiative for action from the previously autonomous subunits to the central staff which is still one step removed from the top executives. Daily operating decisions remain the province of subunit managers but large numbers of rules and procedures constrain individual actions. Uniform salary schedules, formal faculty work load formulae and due process requirements for hiring, firing, student evaluation and other judgments are established. More and more basic information is sought from the subunits: unit costs, faculty activity analysis, student outcomes, student evaluation of faculty, evidence of OSHA or OCR compliance. While still exercising considerable discretion, lower level managers adopt an attitude of "justifying" all their requests to central staff and top executives who are "watchdogs" to insure accountability, efficiency and "rational collective action."[5]

The increasing superstructure of coordination has several consequences. Administrative process becomes valued over substance. Systems, procedures, forms, regulations all require substantial amounts of time, money and staff to maintain, fill out, review, file, write memos about, call meetings to discuss and appoint committees to redesign. Once again, individuals heavily involved in the student-faculty-learning nexus begin to feel isolated and resentful of front office bureaucrats (most of whom have no academic status) who are becoming increasingly involved with academic decisions. Correspondingly, staff members complain about unrealistic, uncooperative, intransigent academics who are unaware of political, demographic and economic realities, who won't modify their irrelevant but sacred cow curricula. Both groups claim that the increasing mistrust and lack of communication result from the bureaucratic, paper-bound straight jacket they are both caught in: the crisis of red tape.

The transition out of red tape bound coordination has been less observed in corporate as well as educational organizations than the previous three transitions. Greiner (1972, p. 44) suggests that phase 5 is one of collaboration with strong interpersonal interdependence. In his view a phase 5 organization is very problem oriented; individuals are organized into teams to resolve important problems. Central staff is reduced through reassignment to inter- or multi-disciplinary teams. Management control systems are simplified and supplemented by frequent conferences of team leaders. Innovation becomes a major value once more with accountability maintained by an overlapping matrix organizational structure and real time information systems. While some research groups and management consulting firms are organized this way, there is little evidence about higher educational institutions in phase 5. (In a later section I hypothesize a second alternative that phase 4 organizations cycle back to become phase 2 "direction" organizations).

While there is little evidence of phase 5 organizations, there is even less evidence of the subsequent phase 5 crisis. Greiner suggests an individual future shock or "psychological saturation" of "employees who grow emotionally and physically exhausted by the intensity of teamwork and the heavy pressure for innovative solutions." He suggests that phase 5 organizations might deal with the crisis of psychological saturation by creating structures which enable employees to alternate intensive work with periods of reflection and reorientation. This is reminiscent of the rationale for academic and administrative sabbatical leaves which have long been a part of higher education (especially for faculty) although the psychological saturation has usually resulted from individual rather than team efforts. At the organizational level the alternation of work and renewal is the focus of recurrent education in several European countries, China and Japan (see Mushkin, 1973 and Steiner, 1972).

I believe this formulation of the organizational life cycle is very relevant for higher education because it gives at least one way of understanding the decade following the Berkeley Free Speech Movement and including the financial crises of higher education. Different institutions in different phases have faced the different crises of leadership, autonomy and control often precipitated by similar forces of student unrest, political intervention and financial resources inadequate for the aspirations of the institution. The last five years has shown in public institutions the crisis of control being resolved in the direction of increasing statewide coordination through statewide systems, coordinating agencies and, recently, 1202 commissions. The same period has shown in many private institutions the crisis of autonomy both within the institution and with respect to state and federal agencies. Those private institutions that survive their identity crisis (and some merge or become public every year) then face rather quickly the crisis of control and this suggests one reason for the increasing interest by private institutions in implementing management, planning, information and control systems.

It is important to realize that Greiner's phase theory is derived from a very rational, problem solving oriented organizational theory. His collaborative phase 5 is essentially task structured with individuals and teams striving to maximize the accomplishment of organizational objectives. This "Rational Man" view of organizational theory has been seriously questioned by Argyris (1973) and others who suggest organizing around human growth and self-actualization. Recently, Torbert (1974) and Lasker and R. Weathersby (1975) have suggested organizational analogs of individual developmental cycles or stages based on the research of Erikson (1959) and Loevinger (1970). These perspectives provide an important link between individual decision making, use of information, concepts of learning and organizational decision making,use of information and concept of learning.[6]

Individual Development

There is no universal agreement on theories of adult development and we should be aware that the use of conceptual structures can produce self-fulfilling prophesies. Nevertheless, understanding the link between characteristic mentalities of adults in higher education and the organizations we call colleges and universities is useful in identifying matches and mismatches in the types of information and decision making structures. Table 2 summarizes Loevinger's (1970) five major stages of adult development. Because in common usage there seems to be a great deal of confusion over what "stages" represent in stage theories of development, it may be well to reiterate several points made by Loevinger (1970):

1. Stages are markers along a continuum and not a series of separate, discrete boxes.
2. Each stage represents a holistic view of personality; the mind set of a person's stage structures reality for that person.
3. Each stage encompasses and incorporates the previous stage--like the continuing growth of a tree.
4. Stages are meant to be descriptive and not normative.
5. There is no necessary relationship between stage and happiness or being "well adjusted". This is a conformist world but there are happy, healthy people at stages 2-5.
6. Stages are not hurdles to be surmounted or tasks to be solved; rather, stages represent a sequence of coping with increasingly deeper problems.
7. If (2) is true, we should recognize that stage theories reflect a stage 4 or 5 level of thinking and people in stages 2 or 3 will probably have different views of the relationships proposed in this paper.

With this quick sketch of the characteristics of adult mentalities, we can now ask: What is the relationship between the phase of organizational development and the stage of individual development which would be conducive to effective administration in higher education? Unfortunately, there is very little research available to answer this question but let me offer some reasoned hypotheses:

1. Organizations in the creative phase would probably be best "managed" (which is almost a contradiction) by highly autonomous individuals who can tolerate ambiguity, work well with other autonomous individuals attracted to new ventures, deal effectively with conflict--especially on the substance of the organization, be objective about themselves, others and the organization, and who can motivate others through personal growth.
2. As an organization transitions to a directive phase with new functional structures, individuals who are at the conscientious stage would probably be the most effective managers. Conscientious individuals who internalize the dream, the vision of a creative organization, have the problem solving and analytical skills to resolve conflict among subgroups and to examine alternatives carefully. At this phase, the formal organizational structure is still pretty loose and conscientious managers are able to view rules as flexible and to maintain an awareness of the individuality of others.

Table 2

Stages of Adult Ego Development and Key Characteristics

Developmental Stage	Key Characteristics	View of Knowledge	View of Information
1. Presocial-Symbiotic-Impulsive			
2. Self Protective	wary, manipulative, exploitive, opportunistic, fearful of being caught; conscious of control, self-protection; externalizes blame	a concrete thing used to get something	power
3. Conformist	Identifies with authority; seeks social approval and acceptance; concerned about appearance, material goods, reputation; seeks to be friendly, helpful; people classified by social group	general information needed for social roles, know how things work, one right answer	comes from authorities; insures conformity, transactions and control systems
4. Conscientious	Becomes more self evaluative; more aware of inner feelings and differentiated feelings and motives for behavior of oneself and others; rules are relative; intense concern for communication; descriptive and normative analysis clearly separated; long term goals and ideals become standard for judgment	proactive, personal skills in problem solving; in each situation there is a best answer; congruence of divergent views sought by rational process	personally relevant, analytical, assay options and consequences
5. Autonomous	Conscious of one's own and others' needs for autonomy; recognizes the complexity of moral issues, people and organizations; willingness to confront conflict and resolve it openly; tolerates ambiguity; objective about one's self and others; motivated by personal growth	personally relevant, insight vs. observation, subjective, emerging from contradictions, useful in transforming oneself	experiential, based on actual environments; synthetic, integrating theory and practice

Sources: Loevinger (1970), Chickering (1974), Lasker (1974); Chickering shows that these stages are analogous to the moral development stages of Kohlberg (1970) and the intellectual stages of development described by Perry (1970) and Piaget (1971).

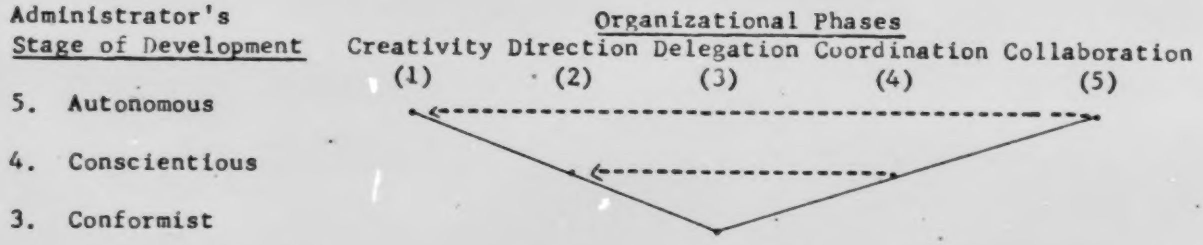
3. Once an organization has adopted a formal, highly delegated structure, effective sub-unit managers are more likely to be conformist. That is, top management usually wants subordinates who will respect their authority, act consistently with the social norms created within the organization, seek approval of their peers and superiors (social disapproval is a major form of control), and who will live by the rules of the organization. These managers should be bright and creative within their spheres of responsibility but the range and hierarchy of responsibility and authority should be very clear and largely unquestioned. If top management is also conformist, then the organization will be highly authoritarian, structured, closed, tight and rarely susceptible to the crisis of control. If top management is conscientious, on the other hand, then the organization is likely to be more open and also more susceptible to disparate evolution leading to the control crisis.

4. If an organization evolves to the coordinated phase, top managers need to be conscientious, assuming more initiative and responsibility for coordinated action requiring better communication and a more differentiated view of others. The process of coordination is usually rational-analytical and conscientious managers view the world that way.

5. Finally, if an organization evolved to the collaborative stage, it seems most likely that autonomous managers would be most effective because of the complexity, fluidity, ambiguity, and highly personal motivation and development inherent in phase 5 organizations.

These hypotheses lead to the following interrelationship figure

Figure 1
Hypothesized Relationship Between
Organizational Phases and Administrator's Stages



This possible relationship in turn leads to one observation and two more hypotheses of organizational evolution. The observation is that as organizations evolve, the most appropriate managerial view of the world may be different for each organizational phase. Many presidential or vice-presidential search committees start with a description of the "ideal candidate" who should be not only a great scholar but also highly principled, autonomous, independent, objective and willing to tolerate an ambiguous, slightly organized anarchy. However, with most colleges and universities in phase 3 and a few in phases 2 and 4 (or so I would judge from personal experience--obviously a subject not only of debate but also of much needed research), autonomous administrators probably offer the most inappropriate type of leadership for most institutions. This suggests that institutions should be aware of their own developmental characteristics and seek administrators whose stage of personal development is congruent with the organization's phase of development--and, conversely, candidates for administrative positions should be aware of the same needed congruity.

The two additional hypotheses are:

1. An organization facing the red tape crisis in the coordination phase is more likely to cycle back to phase 2 (direction) than to evolve into the collaboration phase. As figure 1 suggests, there may be a 2-3-4-2 triangle in which conscientious stage administrators would feel comfortable and effective in operating either in a loosely directive role based on trust, mutual acceptance of organizational and personal goals and flexible rules (phase 2) or in a highly structured, coordinative role based on rational analysis, examining and evaluating alternatives, and seeking a broad overview of individual, organizational and social needs (phase 4).

As external political and economic forces have responded to the perceived crisis of control by requiring public institutions to become more coordinated (either through multi-campus systems or statewide agencies), the cost, cumbersomeness and delays of coordinative planning with the capability of comparing the outputs or effectiveness of alternatives are all readily apparent. This hypothesis would predict that some coordinative structures will evolve back into directive structures as mutual trust develops between institutional and political leaders, as acceptable information is generated which describes the state of the institutions, and as the mutual commitment to the same overarching goals and objectives becomes apparent.

A corollary of this hypothesis is that it is highly unlikely for a phase 4 organization headed by a stage 4 individual to resolve their coordination crisis by becoming highly collaborative without a widespread change of leadership.

2. Collaborative organizations are probably unstable and "fission" into several creative organizations. If Figure 1 accurately depicts effective administrative-organizational relationships, then the collaborative (phase 5) to creative (phase 1) transition is relatively easy or likely because both require individuals with the same personal characteristics. This hypothesis is consistent with the long history of trying to establish collaborative communities and their instabilities (Kanter, 1972) and with Greiner's observations that few organizations ever reach the collaborative phase (because those organizations which become collaborative may rather rapidly fission into creative, phase 1 organization 1).

Decision Structures

Organizational phases may also be characterized by the decision making processes they use. The structure of the decision making and implementation process of an organization reflects both the characteristics of its phase of development and the styles of managers which are characteristic of the manager's stage of development. Meanwhile, the value of information is based on the change in decision or structure caused by the information. In other words, organizational decision structures embody or operationalize the administrator's stage-organizational phase relationship discussed above and provide the context for evaluating alternative information systems.

One schema of institutional decision making[7] is shown in Figure 2. The three rectangles shown in Figure 2 represent the three decision-relevant roles of administrators: formulation of policy options, actual decision and implementation. These roles can be assumed by one individual, or they can be differentiated in a very complex pattern of delegated authority and responsibility, or they can be shared in a loosely structured, fluid, collegial structure, or they can be reorganized weekly--but these three roles must be included in some manner in the organization. More specifically, I argue that there is an evolution of the organization described in Figure 1 and suggest an additional argument for the coordination-direction and collaboration-creative phase transitions.

In the creative phase organizations combine the policy formulation, decision and implementation roles and informally share these roles among the small group of colleagues. While individuals may have functional specialties they are not formally structured. Communication is direct, personal, informal and frequent. The conception-decision-action-evaluation-reconception process is constant, unstructured, occurring over lunch, in the car on the way to a meeting, in the storefront or wherever. For phase 1 organizations, the decision structure is very simple and is shown in Figure 3.

Phase 2 (directive) organizations are functionally organized, communication is formal and structured, authority is usually hierarchial and the decision structure looks like Figure 4. With the operational information filtered through functional managers, top policy makers become increasingly remote and detached from daily activities and increasingly dependent upon implementus for both data and policy recommendations. It is understandable how a crisis of autonomy results from this structure.

Phase 3 organizations (delegated) deal with the autonomy crisis by differentiating the policy implementation structure both horizontally (more specialists) and vertically (more decentralization of authority and responsibility). The resulting decision structure is shown in Figure 5. This graphically visible fragmentation of implementation suggests why the crisis of control usually results from this organizational phase.

Figure 2

Sequence of Institutional Decision Making

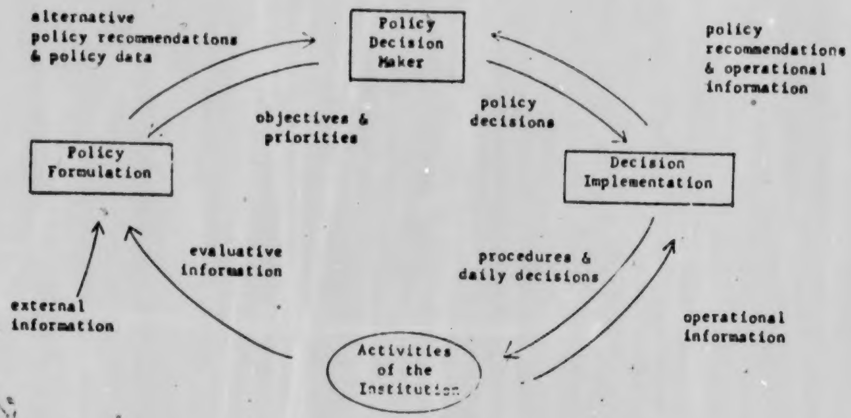


Figure 3

Phase 1 Decision Structure

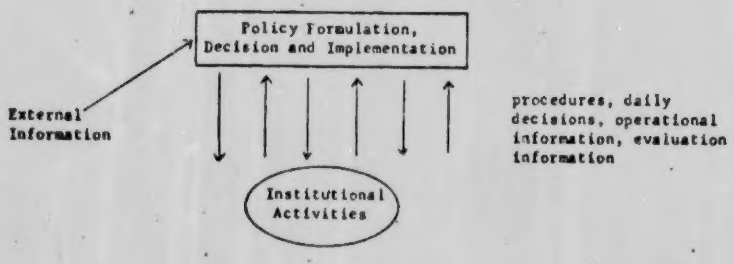


Figure 4

Phase 2 Decision Structure

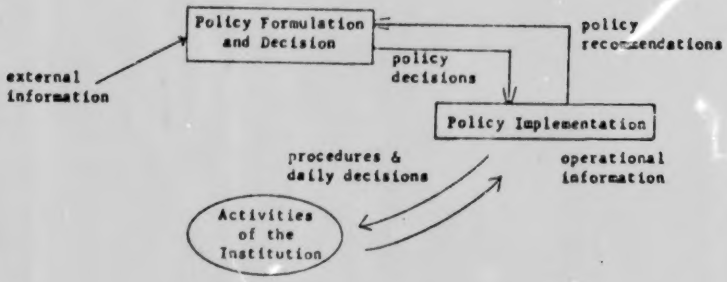
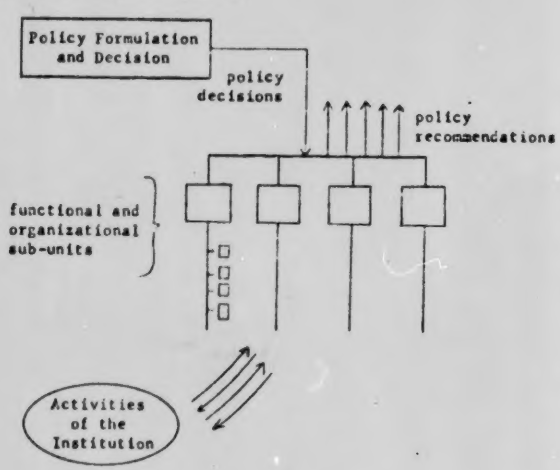


Figure 5

Phase 3 Decision Structure



Phase 4 organizations (coordination) attempt to cut through the mass of overlapping and conflicting functions, procedures, activities and policy recommendations by consolidating implementation units (more powerful vice presidents or provosts) and by establishing a separate evaluation, planning and/or analysis function (see Figure 6). To be effective, this separate policy formulation function must obtain evaluative information about the activities of the institution and about external realities and must interact with the ultimate deciders (president, board of trustees, state legislature or governor or whomever) to make objectives and priorities explicit (at least in-house) and to propose and evaluate alternatives. Often there is an attempt to direct all policy recommendations from functional managers through the policy formulation office. This process of evaluation-analysis of alternatives-policy recommendation-decision-implementation-evaluation becomes increasingly formalized, complex, time consuming and expensive. Institutions and particularly state systems are becoming large bureaucracies heading almost inexorably toward procedural strangulation: the crisis of red tape.

Phase 5 organizations merge the policy formulation and implementation roles into one that is shared by teams of multi-functional specialists organized around particular problem foci. Top management is involved primarily in general direction setting, assignment of individuals to multi-disciplinary teams, resource acquisition, coordination through team leader conferences, and evaluative feedback (see Figure 7). Some research groups, consulting firms, and professional schools seem to be organized this way.

Notice, however, that this phase 5 decision structure is very similar to the phase 2 decision structure shown in Figure 4. If there is significant congruity of objectives and priorities between top and middle management (or between state or board and institutional management), the decision structures of phase 5 and phase 2 become operationally identical. In other words, with common values, mutual trust, and adequate and acceptable information phase 4 organizations evolve into phase 2 organizations.

Another option is for phase 5 organizations to become more collegial with more shared decision making authority, more toleration of individual values and objectives, more acceptance of ambiguity and lack of structure and less adherence to a singular institutional mission. All of these changes blur the distinction between policy formulation, decision and implementation; the top two rectangles in Figure 7 merge into one and the phase 1 decision structure of Figure 3 results.

In summary, if a phase 5 organization remains hierarchial (consistent with the "conscientious" administrator's view of the world) but seeks congruity of values, widely available and acceptable information and personal growth and commitment of its employees, it probably will evolve toward a phase 2 organization. On the other hand, if a phase 5 organization

Figure 6

Phase 4 Decision Structure

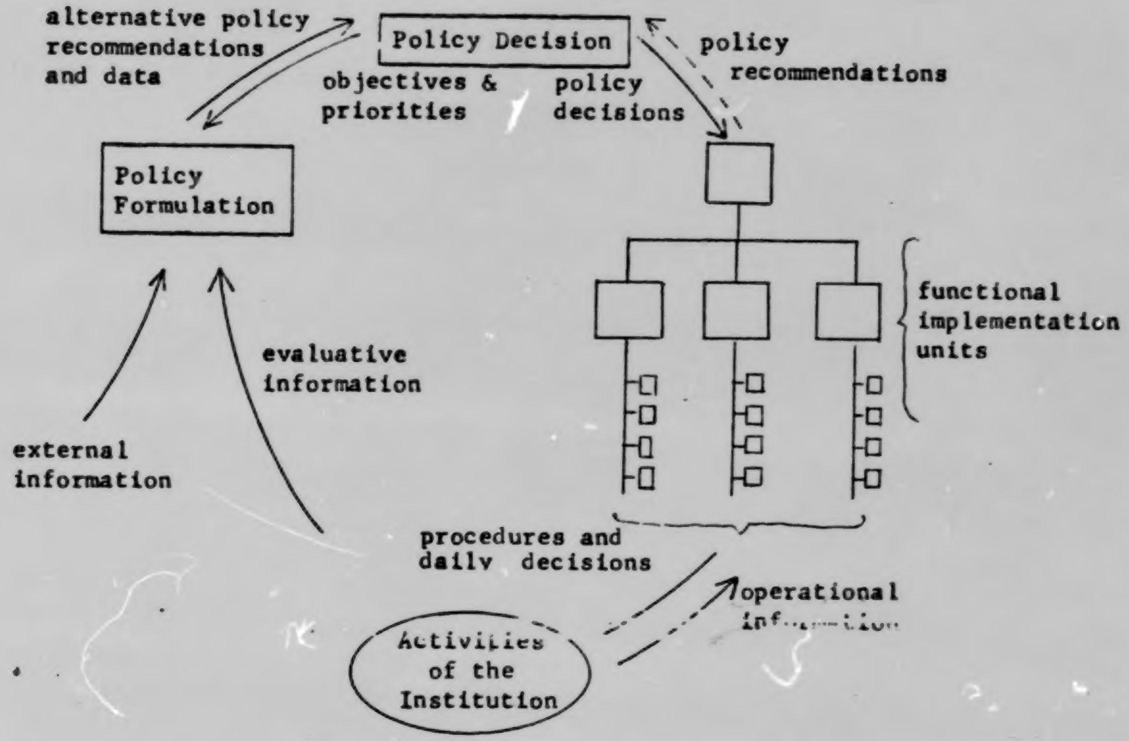
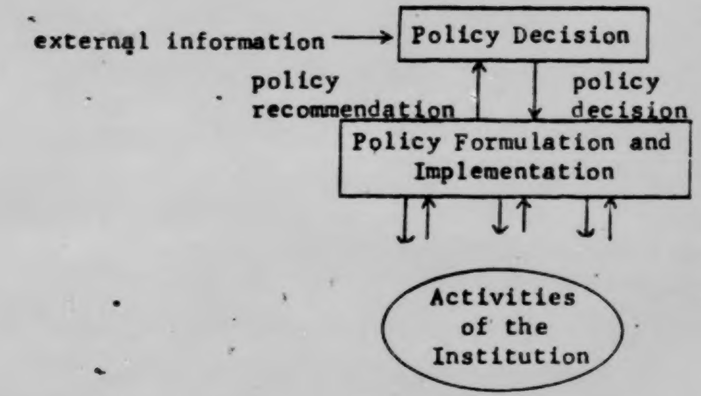


Figure 7

Phase 5 Decision Structure



seeks to become less hierarchial and more collegial, open and multi-faceted (consistent with the "autonomous" administrator's view of the world), it will probably evolve toward or break apart into one or more phase 1 organizations.

Information Systems

The final link in this disucssion is the relationship of information to the various decision structures, organizational phases and administrative stages discussed above. In essence, I argue that the most appropriate information system is determined by these three elements. An administrator should select the appropriate information system according to the phase of organizational development and his or her own managerial perspective of information and decision making--which is largely encapsulated in their stage of personal development.

Mason and Mitcoff (1973) argue that at least implicitly all information systems have five structural characteristics (p. 476):

1. Psychological type of the user (how a person perceives and evaluates the world--or what is knowledge--see Table 3);
2. The type of decision to be made (structured or unstructured, "wicked" problems about which very little is known);
3. Organizational context (policy formulation, decision or implementation-control);
4. Modes of presentation of information (personal interaction or impersonal reports, model results, computer printout); and
5. Philosophy of evidence, inquiry and knowledge (which should be congruent with (1) above).

In this broad view, information systems include empirical as well as intuitive, personal data. Clearly different types of users, problems and organizational contexts will require different information systems (embodying different philosophies) and different modes of presentation.

While this concatenation of "differents" could result in a very large number of alternative information systems, we can draw upon the previous discussion of individual and organizational characteristics to predict the modes of presentation and philosophies of inquiry which would most likely be appropriate for each organizational phase.

Five major philosophies of information systems have been developed by Churchman (1972) and summarized by Mason and Mitroff (1973). Analogous to human development stages, each type of information system includes all of the previous types as subsets or special cases beginning with the simplest Lockean data sets through the most complicated Singerian learning systems. The characteristics of these five types of information systems and a few illustrative examples are listed in Table 3. These types of information systems are then related to organizations in various phases as shown in Table 4.

Table 3

Type of Information System	Characteristics	Examples
1. Lockean	Begin with experience, judgments, basic facts and inductively derive or predict a more general theory, heavy weight given to expert judgment	databanks, accounting, statistics, Delphi Method
2. Leibnitzian	Begin with a basic theory or axioms and deduce general positional truths, collect evidence to validate theory, develop single "best" model and communicate it to (convince) managers	models of student flow, cost allocations
3. Kantian	Begin with at least two alternative basic theories, collect data to verify each theory, present manager with several explicit theories and evidence, allow manager to choose "best" representation of real decision problem	policy analysis, academic program reviews and formulation, simulation of effects of alternatives
4. Hegelian	Begin with at least two completely antithetical theories, collect a single, common data set to evaluate all theories, dialectual confrontation of consciously conflicting models enables manager to formulate a creative synthesis of his/her own theory or model	personal interaction of opposing parties (e.g., law or legislation)
5. Singerian	Continual learning and adaptation, define "wicked" problems in a solvable approximation, use theory and experience to redefine problem in closer approximation, constantly adjust complexity of inquiry to maximize information. (Probably analogous to Argyris' double-loop learning.)	

Sources: Churchman (1971) and Mason and Mitroff (1973).

Table 4
Relationship Between Organizational Phase and Type of Information System

Organizational Phase	Types of Data	Type of Information System
1. Creativity	personal observation, experience, direct feedback	informal, interpersonal, judgmental, synthesis, consensual; (Kantian or Hegelian)
2. Direction	add: functional transactions (accounting, student records)	add: standard format reports of transactions; functional budgets; annual reports; some computerization (Leibnitzian)
3. Delegation	add: more detailed transaction data by operating unit, sources and uses of resources	add: unit cost analysis; seeking comparative data on costs, workload and performance data; computers required (Leibnitzian)
4. Coordination	add: standardized data elements	add: objectives expressed as programs, PPBS attempted; simulation models to evaluate alternatives; program cost analysis; regular data exchange; extensive computerization (Kantian)
5. Collaboration (which may transform to 2 or 1)	add: personal feedback, process feedback	add: flexible output formats; eliminate standard reports, PPBS, standard unit costs analyses; integration of previous decisions with new IS design (Hegelian or Singerian)

add: means the listed characteristics are in addition to the characteristics listed above in the same column.

In the creative phase, information is essentially personal, possibly dialectual, but certainly multiple theories and alternative options are freely considered. Meaningful information is both personally collected and synthesized. In phase 2 (direction) transactions have to be monitored and classes of data collected to insure some control in functional specialties. Accounting, payroll and student record systems become important even if ledger entries are done by hand. This data collection is Lockean but the reporting formats are usually fixed by professional associations or governing boards and the resulting information system is Leibnitzian: the application of one theory of how institutions work and what information is needed.

In phase 3 (delegation) more detailed transaction data are needed and more complicated management accounting and control procedures are used but the basic philosophy is still Leibnitzian. Phase 4 organizations (coordination) need information to examine alternative decisions much more carefully than previous phases and they demand explicit objectives, analysis of consequences of decisions, program budgeting and costing--essentially a Kantian view of information.

Phase 5 organizations that evolve back to phase 2 probably do so with Hegelian, dialectual information systems, which may be predominantly interpersonal based on Leibnitzian or Kantian systems. In this event, the second time around a phase 2 organization is substantively different than the first time because it is able to operate at a much higher level of complexity and communication. Similarly, phase 5 organizations that evolve back to phase 1 are probably quite different than newly formed phase 1 organizations.

Finally, there are (at least conceptually) some phase 5 organizations which develop a Singerian, continuous learning model of behavior which sets them apart qualitatively from other organizations. Loevinger (1970) writes about an integrated stage of adult development which might correspond with the Singerian view of knowledge and learning. Humanist organizational theorists write about openly chosen structures and liberating disciplines (Torbert 1974), theory (MacGregor 1960) and Model II (Argyris 1970); perhaps Table 5 provides one additional way of conceptualizing the interaction of organizational development, individual development, process of decision making and type of information system.

Conclusion

Effective leadership most probably occurs in a congruent environment in which organizational and individual development are synchronous and the decision structure chosen reflects the phase of the organization and the stage of its leaders. Information systems will be most useful when they embody the "world view" of the administrator or the institutional decision making assumptions of the organization. Table 5 suggests one pattern of congruity among institutional phases of development, administrators' stages of development, decision structures and information systems. Individuals responsible for the design of information systems should be cognizant of these interrelationships or their own alternative assumptions if they hope to produce useful information.

Table 5

Summary of Suggested Interrelationships

Organizational Phase (Greiner)	Administrator's Stage Most Appropriate (Loevinger)	Decision Structure Most Appropriate (Weathersby)	Type of Information System Most Appropriate (Churchman)	
			to Organization	to Administrator
1. Creativity	Autonomous	Figure 3	Hegelian	Singerian
2. Direction	Conscientious	Figure 4	Leibnitzian	Kantian
3. Delegation	Conformist	Figure 5	Leibnitzian	Leibnitzian
4. Coordination	Conscientious	Figure 6	Kantian	Kantian
5. Collaborative	Autonomous	Figure 7→3	Singerian	Singerian

FOOTNOTES

1. In 1972 the National Center for Higher Education Management Systems conducted a survey of a large panel of individuals concerned with higher education and asked them to forecast major changes in higher education. This was a five round Delphi survey in which 385 individuals participated, over 200 of whom were college or university administrators; the remainder were Congressmen, governors, state legislators, state administrators, lay-board members, industry leaders, students, faculty, union leaders, foundation officers, and members of the education press. Ninety-four percent of the panel responded to one or more of the five survey rounds. See Huckfeldt (1972).

2. See Weathersby (1974) for a fuller discussion of the study results.

3. See Wyatt and Zeckhauser (1975) for a report of their study of six representative universities and their use of computer based information systems.

4. For a review of the recent research literature in applying organizational behavior to higher education see Balderston (1974).

5. It is well known and frequently observed that "rational" individual decisions do not necessarily lead to "rational" collective decisions but what remains unknown is how top level educational leaders both determine "optional" institutional policies and implement appropriate decisions. See Olson (1965).

6. See Chickering (1974) for an excellent review of the recent developmental psychology literature as it applies to adult development. Hodgkinson (1974) interprets the life cycle theorists implication for higher education administrators and faculty.

7. The decision schema discussed in this section is based on decision theory and is explained more fully in Weathersby (1974). A very interesting sociological perspective on decision making in higher education is given by Helsabeck (1973).

8. One college with which I am familiar has established a non-hierarchial management structure with a rotating "king" for each month and with policy formulation, decisions and implementation primary responsibilities symbolized by blue tags and supporting responsibilities symbolized by yellow tags. Once a week a "tag sale" is held at which the staff bargains for responsibilities (tags) according to individual workloads and capabilities. Unfortunately, this phase 1 decision structure is used by a college undergoing a phase 2 crisis of autonomy and survival with the predictable consequence of very ineffectual decision making. In fact, the delay in decision making may be the death knell of the college.

REFERENCES

1. Argysis, C., Intervention Theory and Method, Reading, Mass: Addison-Wesley, 1970.

2. ________, "Some Limits to Rational Man Organization Theory," Public Administration Review, May/June 1973, pp. 253-267.

3. Balderston, F.E., Managing Today's University, San Francisco: Jossey-Bass, 1974.

4. Chickering, W. W., "The Educational Needs of New Learners: Implication for Liberal Arts Colleges," paper presented at the East Central Colleges Consortium Conference on the New Learners, December, 1974.

5. Churchman, C. W., The Design of Inquiring Systems, New York: Basic Books, 1972.

6. Cohen, M. D. and J. G. March, Leadership and Ambiguity, New York: McGraw-Hill, 1974.

7. Erickson, E., "Identity and the Life Cycle," Psychological Issues, 1959, I(1).

8. Greiner, L. E., "Patterns of Organization Change," Harvard Business Review, May-June, 1967, pp. 119-127.

9. __________, "Evolution and Revolution as Organizations Grow," Harvard Business Review, July-August 1972, pp. 37-46.

10. Helsabeck, R. E., The Compound System: A Conceptual Framework for Effective Decision Making in Colleges, Berkeley: Center for Research and Development in Higher Education, 1973.

11. Hodgkinson, H. L., "Adult Development: Implications for Faculty and Administrators," Educational Record, Fall 1974, pp. 263-274.

12. Huckfeldt, V. E., "A Forecast of Changes in Postsecondary Education," Boulder: NCHEMS at WICHE, 1972.

13. Kanter, R. M., Commitment and Community: Communes and Utopias in Sociological Perspective, Cambridge: Harvard University Press, 1972.

14. Kohlberg, L., "Stages of Moral Development as a Basis for Moral Education," in C. Beck and E. Sullivan (Eds.), Moral Education, University of Toronto Press, 1970.

15. Lasker, H., "Implications of Ego Stage for Adult Education" Mimeograph, Harvard Graduate School of Education, 1974.

16. Lasker, H. and R.E. Weathersby, "Organizational Analogs to Ego Stages: Steps Towards a Theory Relating Individual Functioning with Organizational Environments," Mimeograph, Harvard Graduate School of Education, 1975.

17. Loevinger, J., *Measuring Ego Development*, Vol. 1., San Francisco: Jossey-Bass, 1970.

18. Mason, R. O. and I. I. Mitcoff, "A Program for Research on Management Information Systems," *Management Science*, Vol. 19, No. 5, January 1973, pp. 475-437.

19. McGregor, D., *The Human Side of Enterprise*, New York: McGraw-Hill, 1960.

20. Mushkin, S., *Recurrent Education*, Washington D. C.: Government Printing Office, 1973.

21. Olsen, M., *The Logic of Collective Action: Public Goods and the Theory of Groups*, Cambridge: Harvard University Press, 1965.

22. Perry, W. G., *Forms of Intellectual and Ethical Development in the College Years*. Holt, Rhinehart and Winston. 1970.

23. Piaget, J., "Piaget's Theory," in P. H. Musser (Ed.), *Carmichael's Manual of Child Psychology*, New York: Wiley, 1971.

24. Striner, H., *Continuing Education as a National Capital Investment*, Washington, D. C.: Upjohn Institute, 1972.

25. Torbert, W. R., "Pre-Bureaucratic and Post-Bureaucratic Stages of Organization Development," Mimeograph, Harvard Graduate School of Education, 1974.

26. Weathersby, G. B., "Closing the Loop: New Structures for Planning in American Colleges and Universities," Mimeograph, Harvard Graduate School of Education, 1974.

27. Wyatt, J. and S. Zeckhauser, "Management Information in Universities," Mimeograph, Harvard University Office of Instructional Technology, 1975.

28. Yarmolinsky, A., "Institutional Paralysis," *Daedalus*, Vol. 104, No. 1, pp. 61-67.

TRACK II

Applications Development

Richard Clikeman

Conrad Born

Peter Simmons

Track Coordinator: Donald Tiernan

SOFTWARE PRODUCTS: WHAT'S AVAILABLE THE BENEFITS, AND THE PITFALLS

Richard Clikeman
ICP Quarterly

SOFTWARE PRODUCTS: WHAT'S AVAILABLE THE BENEFITS, AND THE PITFALLS

Richard Clikeman
ICP QUARTERLY

This session reviewed the emergence of the software products industry and argued that cost savings opportunities are available for data processing installations through the use of available products. This presentation also focused on the problems associated with evaluating and implementing application products both from a vendor and user perspective.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

IMPLEMENTING A LIBRARY BOOK ORDERING SYSTEM AT THE UNIVERSITY OF NEW HAMPSHIRE LIBRARY

Conrad C. Born
Systems Analyst
Management Systems Development
University of New Hampshire

The PROBLEM OF ORDERING BOOKS

The most prominent factor that makes ordering books different from, say, ordering cases of canned goods or palates of raw materials for a production process, is the very high volume of processing required per unit cost of the books being ordered. I would venture that it would take less time to order a hundred cases of corned beef hash from a food wholesaler than it would take to order some book that might cost about six dollars.

Furthermore, when the six dollar book arrives, it must be checked more carefully, then cataloged, the proper fund disencumbered and money expended, and probably a notice given to someone of its availability. It must then be shelved or delivered to a branch library. All that corned beef hash seems like a lot less work! For a lot more monetary value received.

It goes further than that. Not only must a book ordering system tie in with cataloging and ultimate storage, providing follow-up in case a book is lost, it must maintain a subsidiary ledger to the University's books. This means some kind of encumbrance and expenditure against an authorized funding, and some approved voucher setup so that the vendor can be paid. It also means that the history of the order must be accessable for a fairly long period of time.

U.N.H LIBRARY'S BOOK ORDERING PROBLEM

The University Library had some problems of its own. The University had grown so that the volume of books being ordered had increased threefold in a seven year period. The existing manual procedure was sound, but the Order Department was faced with the need to increase staff to get caught up. A lead time of several months was not unusual for ordering books. Fund balances were behind actual expenditures or inaccurate. Turnover among the temporary help was high, though they were being pressed into service helping with some of the more complex tasks.

THE DECISION TO PURCHASE A PACKAGED SYSTEM

After a study of the Order Department by the Systems Analyst who was my predecessor, it was decided to purchase the Library Book Ordering System offered by the Baker and Taylor Company in Somerville, New Jersey. Baker and Taylor is a wholesale supply house. The Ordering System, called BATAB, was developed for them by Opner, Associates, and their person who implements and assists in getting BATAB running smoothly was originally was employed by Opner, Associates.

The decision to purchase BATAB was an obvious one. There were several other users all reporting success with the system. The price was very reasonable, since Baker and Taylor had an interest in receiving orders and other data from us on tape rather than hard copy which would have to be converted to machine readable form for entry into their automated system. Our resources were limited. The Library Order Department's need was urgent. Politically within the University Administration, the Data Processing department had a need to provide large scale results with one or two people in a short period of time.

WHAT THE BATAB ORDERING SYSTEM DOES

So much for history. Right now, if you were to request a book be ordered by the U.N.H. Library, you might well be notified of its arrival in a week or two. To make this happen, the Order Department performs a preliminary search procedure to assure the accuracy of your request. This search would also disclose whether the Library or one of the branches might have the book already. A decision to order causes an order clerk to type the order using a flexowriter. From then on the bulk of the record keeping and follow-up is done by the system. I'll list the functions quickly:

Purchase Orders, batched by vendor, ready to mail.
(orders to Baker and Taylor are put on magnetic tape)
Tabulating encumbrances and expenses and computing fund balances.
Checking titles received and producing claims for titles not received.
Checking for duplicate orders.
Assigning a "book number" to each new title so that additional copies can be ordered referencing just the number.
(additional input concerning an order uses the book number as the ID)
Maintaining an Acquisitions History.
Handling expenses & encumbrances and other functions at end of a Fiscal Year.
Providing a semi-annual fund accounting and statistical reports.
Providing Selection lists and reports of New Titles in Agency.
Automatically ordering materials in certain agreed-upon categories. (known as Baker & Taylor's Approval Plan)
Keeping track of cataloging and delivery activity, providing follow-up if required until the book is shelved.
Keeping track of status of open orders both by library and by agency or branch library.

All of this can be done on a multiple library basis. In that case, all of the reports are provided seperately for distribution to each member library in the system. Each member can, of course, have its own branches and agencies.

Needless to say, it would have taken a long time to develop and implement a system like this with the resources that were available to the U.N.H. Department of Administrative Systems. We had one person available.

IMPLEMENTING THE BOOK ORDERING SYSTEM AT THE LIBRARY

This was the easiest part. At least it was easy from the point of view of the Systems Department. Several people from Baker and Taylor arrived with an excellent set of procedural documentation and held training sessions for the personnel of the Order Department. Their manuals were simply worded and to the point; How to order a book; How to make a change to an order; How to cancel an order; etc. We decided to use a Friden Flexowriter to handle all regular orders because of the volume involved. The training and manuals for the use of the Flexowriter were not quite as simple, but the trainees seemed to understand and were producing good input in a few hours. The people from Baker and Taylor seemed to have a clear understanding of the problems of the ordering department. No question went unanswered, no clerical exception processing was ignored. If I were to try to sell this as an ideal system implementation, I would stop right here.

In years past, data processing systems were usually designed around severe limitations in the size and speed of computers. The result sometimes was the preoccupation of the analysts and programmers with the technical part of the design. The human and clerical factors were not dealt with as completely. It seems to be a trend now that systems are designed for the user first, and the machine second. This is a welcome development. Increased computer power at much lower cost makes more complex programs, longer job streams and wider ranges of user options practical. However, in implementing the BATAB system, the majority of the problems we experienced were the limitations of the machine and with the activities of the people that surround the machine.

IMPLEMENTATION IN THE SYSTEMS DEPARTMENT AND COMPUTER CENTER

First I'll give you a brief description of U.N.H. computer facilities and the organization that surrounds the computer. Our machine is an IBM 360 model 50 using OS, MVT and HASP. It is used as a combined facility for academic and administrative use. Academic people use normal batch processing as well as terminals which utilize a limited facility known as CALL/OS. Usage is heavy, with augmentations being performed long after their need became evident. Until recently, disk space for administrative systems was extremely limited.

To control batch administrative processing, several control clerks were hired. Their job was to set up the Job Control for administrative jobs, pull the tapes, schedule the job, and, when run review the output for completeness, separate, burst, decollate and distribute it. To make their work easier and more error free a documentation scheme was developed. All administrative work was to be documented in the same format.

The Administrative Systems Department was a seperate group, located across campus. Our function was to develop systems, document them, and turn them over to the control clerks for routine processing in accordance with a schedule.

In addition to the above geo-political problems, there was a CPU time limit for jobs, a limit in lines that could be printed per job without some sort of exception processing, and a limit to the number of steps permitted per job. Due to disk limitations, administrative source programs were and still are maintained on a tape library. Typical turnaround for a compile and test of a program was one or two days.

Along comes BATAB! Twenty-four step weekly process taking about forty minutes of CPU time with reports regularly exceeding the maximum lines per job that could be spooled on disk using HASP. It took us weeks to get the programs onto our libraries and compiled, and any change was cumbersome to say the least. And when we broke the weekly process down into pieces that would fit within the limits, the sorts were too large to run. Of course we were able to work our way around most of the problems. The point is that our system was less flexible and powerful than the normal IBM 360/50 because of the use to which it was being put. We goofed in not being better prepared with a study of its impact upon our installation.

The process of "cutting over" on to the new system involved a visit from Baker and Taylor people who actually set up the Job Control decks for the first couple of cycles and submitted them to be run. All new orders were prepared for the new system.

Vendor and Fund data was prepared. Here is where we ran into the limitations of our installation again. We could assume only a certain amount of priority. The result was long turnaround, especially considering that a group of Baker and Taylor people were expecting to accomplish several cycles before they left.

The greatest problem we faced occurred after the Baker and Taylor people had left. Although briefed in the running of the BATAB jobs, our control people had trouble setting up the control decks. We were still evolving our documentation scheme, and it appeared to be insufficient to tell the control clerks exactly what files to use for what. The System Analyst had to be involved in production running of the system.

Baker and Taylor had supplied some documentation. In addition to the clerical documentation, which was excellent, there were keypunch instructions and operation instructions for each program. These were confusing our control clerks because they had to break long job streams down into small pieces and look up each piece. The computer operators did not know what program was running since they execute jobs and not programs.

It was at about this time that I became involved in the Book Ordering System. It was evident that the current operating environment had outmoded our scheme of documenting the operating instructions for a computer process. We were documented for a second generation environment! Mount tapes, load the program and press start!

We rethought our philosophy of operating instructions and came up with the following;

Program operating instructions are obsolete. The Job Control runs the operator. He does what the console tells him to do by job. He does not know or care what program is running. He needs to know where to get tapes, how to line up special forms if the job calls for them.

Most console typouts and responses are obsolete, especially if they relate to forms to be mounted or tapes being mounted. The former don't print until later anyway, and the latter is or should be under the control of the Job Control as set up in advance by the control clerks.

If something is wrong with a job, it should abend and be returned to the control clerks for review and correction.

Any problem caused by lack of information in the instructions should the cause of an automatic augmentation of the instructions to minimize the liklihood of a recurrance.

Based on these ideas we improved our process documentation and our problems with normal runs decreased.

MAINTENANCE

Surely everyone has looked at the owner's manual of an automobile and seen the following advice:

> "This operation should be carried out by your friendly dealer who has the special tools . . ."

The Baker and Taylor Book Ordering System is owned and maintained by Baker and Taylor. The programs are copyrighted. Although the user has a copy of the source programs, he does not have any explanation of what the program really does or how it does it. If a problem arises, he is dependent upon the staff at Baker and Taylor for assistance.

As luck would have it, Baker and Taylor is very responsive to cries for help. They are quick to pass along source changes that become necessary to all users. However, their staff is frequently busy installing the system at new locations. Our average response to a telephone request for assistance has been about two days. A written request for changes of a routine nature, such as changing some of the options available to the Library Order Department, would take longer. Generally, it is up to the user to insert the program changes, recompile and test. If the results are funny, the telephone can get a lot of use.

In the event of large or multiple changes, a visit by a Baker and Taylor staff member is usually made. In this case the problem of turnaround and priority becomes a nuisance. In any case, the availability of some part of a systems development staff member's time is mandatory for continued maintainance.

RESULTS FOR THE UNIVERSITY LIBRARY

Given all the problems we have encountered and solved, was the BATAB system a worthwhile solution to the original set of problems of book ordering? Certainly! The Order Department has realized the following benefits;

A drastic reduction in files maintained, reports to be compiled, and bookkeeping.

A reduction in the volume of order entry data to be typed due to the approval plan and the ability to send and receive tapes from Baker and Taylor.

An improvement in turnaround time for orders from Baker and Taylor.

A drastic reduction in clerical time to order books, especially as experienced by the ordering agency.

Much better fund accounting, both in accuracy and timeliness.

A reduction in staff from three librarians to two, from eleven staff members to eight, but an increase in part time personnel from five to seven.

Less sensitivity to peak loads.

Less vulnerability to personnel turnover.

The weekly cost of running the system, supplies, and the availability of personnel to set up, run, and maintain the system appears to be less than the savings in personnel realized in the Order Department. Since the actual volume of orders has risen by some fifteen percent, the per-unit-ordered cost would seem to have dropped slightly.

When one can realize an improvement in service to the ordering agencies and patrons at a reduction in relative cost, things can't be all bad!

TELEPROCESSING CASE STUDY
AN ON-LINE GENERAL STORES SYSTEM

Al J. Dandrea
Senior Systems Analyst
Department of Management Engineering
The Pennsylvania State University

M. M. Bruneau
Systems Analyst
Department of Management Engineering
The Pennsylvania State University

An On-Line General Stores System

This paper is a brief but hopefully informative summary of the more important experiences in developing an on-line transaction oriented general stores inventory system. Special attention is given to those topics which relate to the on-line aspects of the design problem with little attention given to the more standard batch processing portions.

A. Why an On-Line System?

One of the first problems in developing this system was to recognize that we needed an on-line capability. Some of the factors which both suggested and justified considering an on-line system were (a) the need for computer pricing to appear on the original requisition, (b) the need to have quick handling of certain exceptions (e.g. large price changes or negative inventory conditions), and (c) the desire for the buyers to have the most current inventory status when ordering by phone or checking the supply of critical items. A brief discussion of the need for computer pricing of the original requisition might be useful at this time. A signature of authorization is required on each requisition, but, with some 5,000 requisitions a month, it seemed very inefficient to be attaching computer printed pricing data to each of 5 copies of the requisition. Thus a method of having the computer type the pricing directly onto the original requisition was evolved. This will be discussed further under Terminal Selection.

B. Software Considerations

This system had to be operational on a 370/145 facility so the three following alternatives were considered for the on-line portion of the system.

1) Procure a vendor-supplied application package.
2) Develop an in-house system using the time-sharing option (TSO). or
3) Develop an in-house system using real-time processing (CICS).

Although available software packages for general inventory would have had the advantage of saving a lot of in-house program development, several circumstances of this system discouraged any serious inquiries. Some of the circumstances included (a) the requirement to eventually link the inventory system to the current accounting system for direct billing, (b) the immediate need to have current prices and totals printed on command directly onto the original source document, and (c) the requirement that any on-line application package be completely compatible with the OS/MVT and CICS environment that existed at that time.

The use of time-sharing not only for development, but also for the operational system seems to have the most advantages. The interactive debug features and quick turnaround as well as being able to use a standard language like COBOL or PL-I all would have resulted in significant reduction in development effort. Perhaps even more beneficial would be the almost total control of the operational system that could be given to the General Stores department. This control could have extended to ability to output various weekly and monthly reports without need for extensive scheduling and control efforts needed at the central batch facility. Extensive experience with other operational time-shared systems on another facility on campus has shown that the operating departments can indeed assume this much control and responsibility with proper system design and documentation. Nonetheless, time sharing had to be ruled out because this initial development effort using TSO was found to cause a marked impairment of routine production in the OS/MVT environment. If a virtual (VS) environment had existed at that time, this system likely would have been developed using time-sharing.

The decision to use a real-time control system (CICS) to develop the general stores application necessitated special programmer training, batch development at the computer facility, and design of a hybrid operation (real-time input and edit with batch output of reports). However, in learning the use of the CICS control statements, the programmer-analyst also uncovered several desirable features of BMS (basic mapping support), a support feature within CICS. BMS enables high level control of such things as highlighting, indentations, initializing data fields, and automatic positioning of the cursor. As we will see later, these features are very important for an easy-to-use real-time system.

C. Terminal Selection

Once a real-time software control was chosen, the use of a CRT type display terminal was almost evident because of its proven history of easy operator training and use in real-time business applications. However, the need to have computer pricing appear on the original requisition caused some rethinking about the best terminal for this job. The answer to both of our terminal needs was the IBM 3270 display terminal with 3286 printer attached. The printer is a 66 character per second impact type which can be configured for control from the 3270 keyboard, but with its own format. This enables complete data entry and editing on the 3270, and when everything is verified the requisition is inserted in the typewriter-like platen of the 3286 and on command the unit price and totals columns are printed.

D. Display Techniques

Perhaps the most challenging aspect of this on-line approach as opposed to a batch system was designing data entry, error message, and correction displays which would be reliably usable by the average inventory clerk. The need for at least 9 different data entry transaction types made for a particularly complex development. Each entry transaction was carefully studied and cross-correlated with the others to see patterns of similarities or contrast. This revealed that 7 of the transaction types could be grouped together at least for a common index for starting purposes. This index is shown in figure 3 which is the actual display format that the clerk could invoke. She chooses which transaction she wishes to perform and types in its 3 letter code and the item number on the lower left of the screen. The next display she sees is arrayed especially for input of only pertinent data from the source document. Once the data is entered she next sees an error display or a verification display. Figure 4 is an example of an error display (actually a warning in this case) for a receipt entry. Note at the bottom line a "YES" has been typed indicating that data has been verified and the buyer consulted and the price change should take effect. The resultant verification display is shown in figure 5. Underlining is used in these figures wherever brightness highlighting occurs on the actual display. This serves to focus attention on special messages or data.

The most voluminous and distinct transaction type is the requisition so it was a natural for a special transaction just for requisitions. For each data line on the source document just 3 data elements are entered. These are the quantity, the order unit, and the item number. If a line on the requisition is blank it must be entered blank on the display so the final priced print-out will line up correctly. Figure 6 is a sample of the error display that could result if a requisition has been improperly entered. Note again that underlined passages would be highlighted on the actual display. The item description comes from the computer files and should be used in verification. Figure 7 is the verification display after errors have been corrected. If final inspection shows all to be correct, the original requisition set is placed in the printer and special command keys both start the printing action for unit price and total and update the inventory files. Figure 8 is a list of some typical error messages that might appear in highlighted form on various transactions.

Because of its infrequent use and special data preparation, the order unit change was developed as a separate transaction.

Figure 9 shows the various steps needed to convert the existing inventory from one unit to another (e.g. gallons to quarts).

E. Special Considerations

As noted earlier this system has both real-time input and update and batch report writing. So besides the usual systems and program documentation, special time and effort had to be given to carefully writing operating manuals for the clerical staff that use the system. These manuals serve as back-up to hands-on training sessions which were conducted over a 10 day period during bring-up.

The problem of a "head-crash" on the disc during operation seemed to pose a difficult problem for this system. The data could be restored as of 8 AM, but what about reentering all those requisitions that have already been priced and sent to the customers? The answer was to generate a transaction log for each transaction for which there was a file update. These logs could then be recycled through the computer should a crash occur and files have to be restored to 8 AM status. As figure 10 mentions, a byproduct of this recovery process is an audit trail which is listed weekly showing, in sorted order, each requisition and its detail along with special updates (e.g. receipts, inventory adjustments, etc.).

One other difficulty that occurs when working with a real-time system comes when you want to test a modification to a display edit or update. A special test version of CICS had to be brought up so that any drastic programming bugs would not interfere with the production users of CICS. This often necessitates special production arrangements or off-hour testing if you are operating under a OS/MVT environment. A VS environment is much more accommodating.

WHY AN ON-LINE GENERAL STORES SYSTEM?

1. SAME DAY UPDATE OF INVENTORY STATUS.
2. TERMINAL PRICING OF ORIGINAL REQUISITION.
3. IN-HOUSE HANDLING OF EXCEPTIONS.
    A. UNIT PRICE CHANGES.
    B. NEGATIVE INVENTORY ITEMS.

Figure 1

ON-LINE SOFTWARE CONSIDERATIONS

1. COMMERCIAL SOFTWARE PACKAGES.
    A. CUSTOM FEATURES NEEDED
    B. FUTURE EXPANSION
2. TIME-SHARING OPTION (TSO).
    A. MANY ADVANTAGES
    B. BAD TIMING
3. CICS (IBM REAL-TIME SYSTEM).
    A. PROVEN PERFORMANCE VS COSTLY DEVELOPMENT
    B. BASIC MAPPING SUPPORT (BMS)

Figure 2

```
                    GENERAL STORES INVENTORY FILE UPDATE

          IN ALL TRANSACTIONS, NNNNNN REPRESENTS A 6-DIGIT ITEM NUMBER

1.  NEW NNNNNN - CREATE AN INVENTORY RECORD FOR A NEW ITEM

2.  CHG NNNNNN - CHANGE ITEM DESCRIPTION, REORDER POINT, MAXIMUM, VENDOR, PO #,
                 RECEIPT DATE, AND/OR MARKUP

3.  DEL NNNNNN - DELETE AN ITEM FROM THE INVENTORY FILE

4.  CRM NNNNNN - CREDIT MEMO ADJUSTMENT (CHANGE INVENTORY QUANTITY)

5.  QTY NNNNNN - INVENTORY QUANTITY ADJUSTMENT

6.  VAL NNNNNN - VALUE ADJUSTMENT (ADJUST TOTAL VALUE OF INVENTORY)

7.  REC NNNNNN - PROCESS RECEIPT OF AN ITEM

8.  END          END TRANSACTION WITHOUT PROCESSING ANY DATA

*** REC  704200
```

Figure 3

```
                    GENERAL STORES INVENTORY RECEIPT

        ITEM # 407650

        DESCRIPTION:  TUBE-12AX7A
                                                          ERRORS
    1.  INVENTORY QUANTITY INCREMENT          10

    2.  INVENTORY VALUE INCREMENT          21.00

    3.  PURCHASE ORDER NUMBER           74113

    4.  RECEIPT DATE                    11/25/74

    5.  VENDOR NAME                     ALVO ELECTRONICS

    **** WARNING - UNIT PRICE CHANGES BY MORE THAN 25 PER CENT ****

            OLD PRICE:  1.65        NEW PRICE:  2.31

        **** TO FORCE PRICE CHANGE, ENTER YES:  YES ****
```

Figure 4

```
                    GENERAL STORES INVENTORY RECEIPT

THE FOLLOWING CHANGES HAVE BEEN MADE TO THE GENERAL STORES INVENTORY FILE FOR:

     ITEM # 407650

     DESCRIPTION:  TUBE-12AX7A

1.   INVENTORY QUANTITY      OLD      0.00   NEW:    10.00   INCREMENT:   10.00

2.   INVENTORY VALUE         OLD      0.00   NEW:    23.10   INCREMENT:   23.10
     A.  CALCULATED PRICE    OLD      1.65   NEW:     2.31

3.   PURCHASE ORDER #        OLD  69573      NEW:  74113

4.   RECEIPT DATE            OLD  4/16/73    NEW  11/25/74

5.   VENDOR NAME             OLD  R & P DIST.
                             NEW  ALVO ELECTRONICS
```

Figure 5

```
                DISPLAY FORMAT FOR REQUISITION WITH ERRORS

REQUISITION NUMBER:  780195

LINE    QTY    UNIT    ITEM #    DESCRIPTION                      UNIT PRICE TOTAL

 01      40    EA      312960    LAMPS-60W A19 IF

 02

 03      19    QT      279800    NEGATIVE INVENTORY RESULTS

 04       5    M       714700    PAPER-BOND 20# 8½ x 11 WHITE

 05      29    BOX     717580    INVALID ORDER UNIT

 06       3    ROLL    518650    INVALID ID : ITEM NOT ON FILE

 07
  .
  :
  .
 18

    ****  ERRORS
```

Figure 6

DISPLAY FORMAT FOR REQUISITION (ERRORS CORRECTED)

```
REQUISITION NUMBER:  780195

LINE    QTY    UNIT    ITEM #    DESCRIPTION                     UNIT PRICE  TOTAL

 01      40    EA      312960    LAMPS-60W                             .185    7.40

 02

 03      10    QT      279800    SPACKLING                             .752    7.52

 04       5    M       714700    PAPER-BOND 20# 8½ x 11 WHITE         2.803   14.02

 05      29    DOZ     717580    PENCILS 2½ WITH ERASERS               .280    8.12

 06       3    ROLL    518560    FIRST AID-TAPE ADHESIVE 1"            .372    1.12

 07
  .
  .
  .
 18

    ****   TOTAL:                                                             38.18
```

Figure 7

TYPICAL ERROR MESSAGES (HIGH-LIGHTED)

- INVALID REQ #
- INVALID ITEM NUMBER
- ITEM NUMBER IS NOT NUMERIC
- ITEM HAS BEEN DELETED
- INVALID ORDER UNIT
- QUANTITY OUT OF RANGE
- DECIMAL QUANTITY INVALID
- VALUE = ZERO, QTY NOT = ZERO
- RESULT IS NEGATIVE QUANTITY

Figure 8

GENERAL STORES INVENTORY ORDER UNIT CHANGE

ITEM # 238210

DESCRIPTION: LINSEED OIL - RAW

REQUIRED INFORMATION:

1. ORDER UNIT OLD: GAL NEW: QT
2. UNIT-TO-UNIT CONVERSION 4 NEW ORDER UNIT(S) = 1 OLD ORDER UNIT(S)
3. RE-ORDER POINT OLD: 10 NEW: 40
4. MAXIMUM INVENTORY OLD: 100 NEW: 400

OPTIONAL INFORMATION:

1. NEW ITEM DESCRIPTION:
2. MARKUP PERCENTAGE OLD: 12 NEW:

Figure 9

RECOVERY CONSIDERATIONS

1. MANUAL REENTRY OF DATA IMPRACTICAL.

2. EACH TRANSACTION GENERATES A LOGGING RECORD.

3. RECOVERY PROCEDURE
    - A. STARTS WITH 08:00 AM FILES
    - B. AUTOMATICALLY REPROCESSES USING LOGGING RECORDS

4. TRANSACTION AUDIT TRAIL - A BY-PRODUCT.

Figure 10

IMPLEMENTING AN ON-LINE DATA BASE SYSTEM FOR PERSONNEL RECORDS

Pat Urban
Systems Analyst
Management Information & Computing
University of Vermont

IMPLEMENTING AN ON-LINE DATA BASE SYSTEM FOR PERSONNEL RECORDS

The University of Vermont has been maintaining an on-line data base for Personnel records since the spring of '73. The Personnel System, designed in house within the framework of data base management software, supports the user office with the following features:

- an on-line update network
- a general purpose query facility
- a general purpose report generator
- on-line retrieval of data in source document format
- batch programs to support reporting requirements

The system was designed to function in the future as an integral part of a Personnel/Payroll/Distribution System. Therefore, all the financial data elements that would be required to build a data base for such a system were included in the data definition dictionary. The source documents for data collection were also designed to reflect the three-way organization of the data base. Since a central input group to support the three areas would be formed and physically located, with the CRT's, in the Personnel office and since that office was to be the point of collection of all data, the Personnel portion of the system was designated first to become operational. To date, it is only the Personnel System that operates in production mode and so it is that system with which this discussion is primarily concerned.

Search For An Effective DBMS

It was purposely intended that the data base management software in use at this institution has not been referred to by name. The sole reason is that the problems here discussed deal with the implementation of an on-line application data base and are not peculiar to any particular brand of data base management software. The decisive forces which determine the acquisition of software products are the considerations of administrators who must be influenced by the advice of systems programmers, financial constraints and institutional planning policies. To deal with those is beyond the scope of this paper (and also beyond the scope of this analyst).

It suffices to say, strictly from the aspect of applications development, that a powerful file inquiry service and a report generator facility were important factors in the choice of data base management software.

DBMS vs. Application

For the sake of clarification the distinction must be drawn between those features and capabilities that are functions of the data base management software and those that are directly attributable to the Personnel application programs developed under it. The DBMS itself provides the following:

- data management services which provide for definition of the data element dictionary and creation of the data base from variable-length records
- services which provide for the access and updating of elements in either batch or terminal application programs

via standard calling procedures, for example,

Call RPLDATA using ....(parameter list)....

which replaces an element value in the data base

- terminal services, also invoked via calling procedures in terminal application programs, which provide for the sending and receiving of screen page formats, the transmitting of data to the screen and the retrieval of responses, for example,

  Call TGET using ....(parameter list)....

  which retrieves data from a block on the screen
- a generalized on-line file inquiry, probably the most powerful feature of the DBMS, which allows a user to access and order any collection of data and/or perform a variety of arithmetic functions on data (see Fig. 0)
- a generalized report writer feature which operates on-line to allow a user to catalogue a report format containing specific elements and repeatedly retrieve it via different selection criteria (see appendix A for sample report writer)

The Personnel application programs themselves, designed and written in house, include:

- a terminal update program, written in a network of COBOL modules (approx. 250 of them, each no larger than 2K) which provides for on-line entry of four data collection documents
- batch programs tailored to satisfy regularly scheduled reporting and statistical requirements
- file maintenance programs

- a general print program which allows a user to specify at will up to fifteen elements to be printed via any selection criteria, thus satisfying ninety-nine percent of the 'one-shot' requests received (see appendix B for a sample report)

Note: There is a basic difference between this general print program and the report generator provided by the DBMS. The report generator produces a fixed format of elements that has been cataloged ahead of time giving the user the freedom to arrange spacing and headings in any manner he wishes. With the batch print program any group of data items may be requested with no prior catalogue of format, but they will be reported with predetermined spacing and heading information: This program has proven invaluable for filling those 'one-shot' requests for reports.

User Needs Requiring An Updated System

The Personnel Office, as the second largest user of Administrative Systems and having a justifiable need for immediate access to data, was one of the first candidates for conversion to DBMS. In order to demonstrate this need, it must be pointed out that the types of data maintained by Personnel Records at this university might, in some other institution, be collected by another office, such as that of Institutional Studies or the academic dean. The following are some of those data groups:

- all faculty data necessary for consideration of reappointments, leaves, sabbaticals, and tenure

- all data necessary for consideration of administrative reappointments
- data elements required to accomplish federal reporting such as Equal Opportunity, Affirmative Action and National Science Foundation
- all benefits information such as TIAA, health insurance plans, etc.
- educational data, including degrees, publications, committees, certifications and licenses
- job status information such as job titles, department assignments, employee type and salary
- personal data, including addresses and dependents

From the range of data collected and maintained, we can understand the widespread line of services provided by the Personnel office for the University community.

A. Inadequacies In the Old System

Prior to the development of the Personnel data base, a traditionally organized index sequential file was maintained. It was keyed by social security number and functioned adequately enough for the time period for which it was developed. The input medium was paper tape which was punched using codes to identify each field to be updated. The paper tape was punched to cards and the cards, in turn, were edited and put on mag tape. At this point an update program was run which produced an audit trail of error transactions. The whole process had then to be repeated to correct the error transactions. It was not

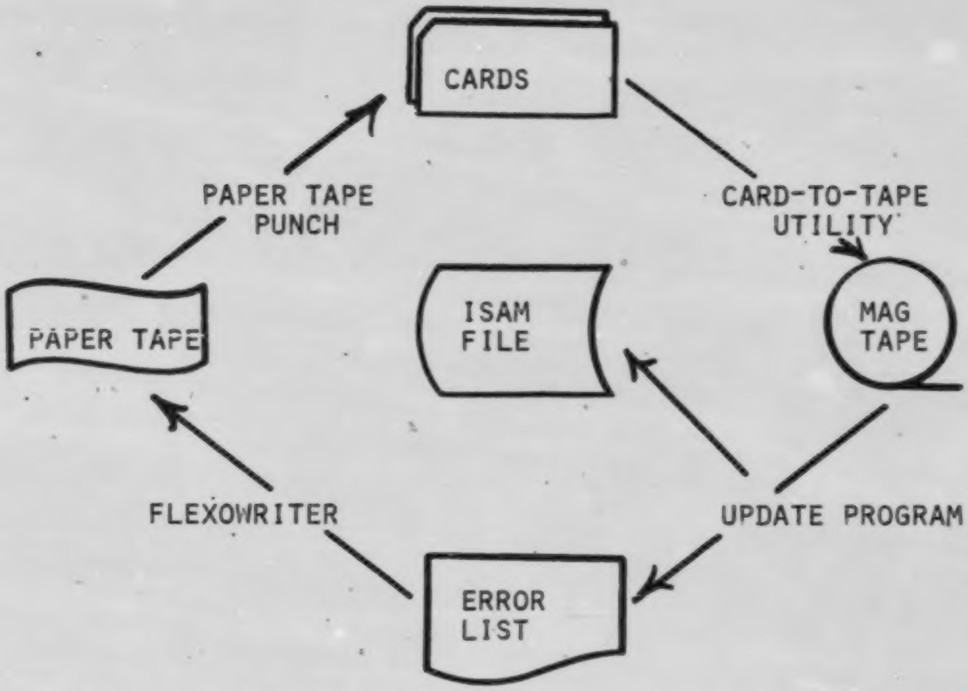


Fig. 1 - Update flow for traditional file structure.

unusual for an entire week to pass for a complete set of transactions to be turned around. Thus, as in Fig. 1, we had an update circle that was difficult to break out of. This whole process resulted in data that never really seemed to be up-to-date.

B. Demands By the University Community

Because of the nature of the data maintained by the Personnel office, there are numerous requests by the University community for services and data which, in the old system, often seemed to require an alteration to an existing program or a wasteful 'one-shot' program written for some specific retrieval.

As a result, there was a high incidence of 'maintenance' programming being done. To compound the problem, there was a large volume of paper being generated in case some bit of data might be required on short notice. Periodically, the entire file would be listed in the interest of supporting rapid inquiry or file lookup. This eliminated waiting for a printout that might, if rushed, take a half day to produce. Needless to say, the worth of dumping three to six thousand records to provide the data possibly needed from one such record is questionable. On the other hand, there was the argument that the time and effort necessary to produce a specific printout to answer an inquiry might negate the worth of answering the question at all. On-line access to an up-to-date file seemed invaluable in the face of these types of problems.

C. Federal Reporting Requirements

In addition to those user problems already cited, there were the ever increasing reporting requirements imposed by the Federal government. For example, statistics for an Equal Opportunity Report had to be calculated by a tailored batch program. Requirements for Affirmative Action and other reporting were demanding data elements that were not even being collected.

D. Desire of Personnel To Increase Data Collection

There was interest expressed by Personnel management to carry data that to-date had only been maintained in paper files or had not been collected at all. The types of data involved e.g. benefits, publications, committees, historical data, were, at best, cumbersome to handle in traditional file structures with traditional input devices.

E. Interaction With the Payroll Office

The Personnel Action Form, the document that transmits any change in job status, title or salary from the initiating department to the responsible administrative offices, was a constant source of confusion in the old system. These "PA's" were ordinarily routed first to Personnel for updating of records and then on to the Payroll office for processing. In addition to the duplication of effort involved in maintaining two files for the same employees , there was a high incidence of discrepancy in the data maintained by the two offices. Forms might mistakenly be sent directly to the Payroll office, bypassing the Personnel update process. It was also possible for documents to get held up in a backlog of updating in Personnel and never make it to Payroll by the required date. It was recognized that one employee file might eliminate the redundancies and discrepancies that plagued the separate files.

Problems of Data Base Implementation

A. File Design

Perhaps the most difficult aspect of launching the Personnel data base effort was design of the file (our first attempt) within the framework of our DBMS and creation of the data element dictionary. The data base is physically a collection of variable-length records, each of which is composed of one or more fixed-length segments of data elements. These data elements are usually grouped logically within segments by the way in which they will be retrieved applicationwise. Each segment may occur optionally from zero to many times. The option also exists to make a segment of data mandatory for at least one occurrence and/or limited to one occurrence.

For example (refer to Fig. 2), a faculty member may have three degrees and so have three occurrences of the degree information segment (#025). An accountant, on the other hand, may have only one degree but also have a CPA, thereby necessitating one occurrence of the degree segment (#025) and one occurrence of the certification segment (#030). A clerk may have no degree segment at all. All employee types are required to have the general employee data segment (#001) and a work address segment (#003).

In order to work with the data elements most efficiently, worksheets were devised that would give some detail on each element proposed in order to facilitate proper placement of the item in the data element dictionary. The worksheet included tentative segment number, the data name it would carry on the file, a brief description, whether it was a new or current element, its COBOL picture, for what it was needed, its collection point and its updating point (see Fig. 3). These worksheets were utilized for several weeks prior to the first build of the dictionary. Even then a large amount of reworking became necessary, but the use of the worksheets definitely held the changes to a minimum.

Constant communication between analyst and user is critical at the data base design stage. Conversion from a traditional file structure will at least form a skeleton of necessary data items. From that point those elements must be determined that will satisfy new user needs and support the more sophisticated system capabilities that are available. A user's familiarity with the daily operation of his office and his cognizance of future requirements are essential to the success of the design process. It is only through communication that an analyst can design a data base system that will enhance a user's capabilities and simplify day-to-day procedures. It will be the degree of a user's satisfaction with the new system, and his willingness to use it, that will make it successful or not. Also, if the new system is the first effort in data base conversion in an installation, it may well determine the acceptance or rejection of the whole data base concept. Poor design because of inadequate communication of specifications will, of course, produce a system that will not function to suit the user needs. If this occurs in the new environment, the blame may erroneously be placed on the entire data base software.

Communication becomes more complex when the time arrives for users who will be relinquishing seperate files to interact on matters that cross offices. Individual users find they must often adopt a unanimous data definition to replace multiple separate ones or compromise on a single new procedure. They must also realize that affecting an item of data in the new environment may have implications for other offices using the same data. Hopefully, the majority of issues raised during file consolidation and new design can be resolved on a discussion and compromise basis among analysts and user offices. If such is not the case, it becomes mandatory for a higher level of management to make a decision based on alternatives presented. If this high-level administrative commitment is not present, problems will never be resolved as each office refuses to give up ground to the others. It cannot be stressed too strongly that management commitment, together with the desire of users for data base capabilities, are essential ingredients of a successfully integrated data base system.

The following are examples of some of the smaller problems and issues that became points of discussion between the Personnel and Payroll offices:

- Names were kept in different formats by the two offices, each for their own purposes. It was not unusual to find the same social security number with a considerably different name on each of the files. (Neither was it unusual to find the same name with discrepancies in social security number between the two files.)
- Many definitions came under scrutiny, such as that of 'salary'. If an employee was hired for the last six months of a year at $6,000, was his annual salary at $6,000 (as Personnel advocated) or did he make an annualized salary of $12,000 (as maintained by Payroll)?

- Benefit plans kept by Personnel had to match up with payroll deductions maintained by the Payroll office.
- Personnel did not consider temporary workers to be employees of the University (as they received no benefits) and consequently did not maintain any record of them. Of course the Payroll office needed basic data on them just for the paying process. Yet, the Personnel staff did not want records of temporary employees 'cluttering up' the file and diluting their statistical counts.
- Addresses were another point of contention. Payroll was interested in permanent address (for W2's) and various addresses for check mailing. Personnel was only interested in maintaining a permanent address and one's campus location.

We also have an example of user interaction on a grander scale. It became necessary for the Personnel/Payroll/Distribution data base users to meet with those users of the Student Records data base in the interest of consolidating peripheral files common to both areas, such as the building code table and the department code table. Our efforts at consolidation have begun to make headway as we are still maintaining two department code tables but finally have arrived at one building code file. Again, the intervention of higher management may become necessary to arbitrate unresolvable differences between user offices.

Communication is not reserved strictly for use among users or between analysts and users. It must also be pointed out that communication between programmers and analysts working on the same data base application is critical. Centralization of the responsibility for the data element dictionary and the project as a whole is a step in the direction of cohesiveness. Implementation of structured programming techniques, such as cataloging of data sets in source statement libraries, the chief-programmer team concept, and modular programming are invaluable tools in a data base environment.

B. Data Collection

The data base is on-line during regular working hours when all updating of personnel records is accomplished. There are four basic documents for data collection, designed specifically for use with on-line entry. They are:

- DATA FORM - a basic biographical document usually completed only once at the beginning of employment
- PERSONNEL ACTION FORM - the document used by the department to initiate a change in job status, title or salary
- BENEFIT FORM - used to capture all benefit, deduction and tax data (includes a W4 image on the form)
- SALARY DISTRIBUTION FORM - collects all salary distribution account data from the departments

(see Appendix C-F for sample forms)

Because it is desirable with on-line entry to leave some sort of audit trail of transactions entered, the forms were designed using the concept of the 'turnaround document'. There are two lines, one shaded and one not, for every data item which is captured (see Fig. 4). When a form is completed for the first time, data is hand-written in the non-shaded area and then submitted for on-line entry (Fig. 4A). On the evening shift a batch program is run which reprints, on the shaded line, all data on file for those documents entered that day (Fig. 4B). When a change must be made, only those fields to be altered are updated in the area directly underneath the current data for those fields. Only those items to be changed need be entered, not the entire form (Fig. 4C). Changes made are entered via CRT and are reflected when the new document is reprinted in batch (Fig. 4D). Steps 4C and 4D are repeated for all future changes and corrections. The documents are serially numbered to insure uniqueness and security.

Design of the four documents was a joint effort of the Personnel, Payroll, and Distribution offices to capture all the data elements necessary to support the three areas. Ultimately the problem of which office would update and maintain what data elements arose. It became obvious that to try to divide the responsibility for updating would only serve to fragment the data base and undermine the whole centralization effort. The concept of a central input group, composed of staff from both the Personnel and Payroll offices was formed. The Personnel and Payroll functions were physically merged into a single combined administrative office. Now all document flow could be channeled to one collection point and all data entry could be accomplished in one location.

This reorganization of administrative offices which reflected the organizational structure of the data base did not occur overnight and did not happen without some repercussions. When we begin to centralize data files and eliminate duplication of effort, there are bound to be political and organizational overtones. Once more we realize the crucial role of administrative support for the data base environment.

C. Time Projections

After implementation of the data base software, being in the position of having to project a target date for the first application system would not be a good spot to find oneself in. Missing deadlines seems to be a hazard of data base development. Probably the most common cause of doing so is either overlooking or underestimating the learning curve involved. Bringing up a data base is like nothing one has ever done before in data processing. The formal education of computer staff in data base techniques can only be supplemented by trial-and-error situations. This is particularly true as data base systems are just emerging into widespread usage. The size of the learning curve is directly proportionate to the size of the first development effort. When the staff of an entire installation is all experimenting with new methods at the same time, the curve is bound to be broader as there is no expertise on which to draw. And if everyone is simultaneously busy with the learning process, there is obviously going to be a lack of productivity in the office. If, at the same time, a large project is being affected by these trial-and-error methods, recovery from error will be more complex and the learning curve will heighten. If a few reliable programmer/analysts could attain expertise by initially implementing a small application, their experience would provide the education for those assigned to subsequent projects. The initial effort would involve a smaller learning curve which would then decrease in size as each new application were undertaken. A small application implemented in a reasonable amount of time will also lend credibility to the data base concept. Larger projects could be begun with the living proof of a working application to lend confidence to the effort.

Neither must the learning curve of users be overlooked. Education in data base concepts must be very thoroughly accomplished, especially when on-line capabilities are being introduced for the first time. There is also the extra coordination to be considered when multiple offices are to be sharing a data base.

It was approximately eighteen (long) months from the receipt of data base software to operation of the Personnel system in production mode. This is somewhat more time than our initial projection (or even our second projection). As shown in Fig. 6, several phases of the project were worked on simultaneously. The overlapping of the design and programming stages indicates that some redesign was necessary once coding of the update network began. User training and on-line test were begun as each new document was completed in the on-line network. The graph illustrates what actually happened in terms of elapsed months of effort. It must be said that future time estimates for development of other data base applications have been much more accurate, as we now have an experience on which to base them.

D. Security

When terminals are available in user offices with the capability of direct access to any data in any file, security features of the data base software become a prime consideration. The Personnel/Payroll/Distribution data base has four levels of security clearance, each with its own identifying password, that allow different levels of access and modification to data elements in both on-line and batch modes. For example:

- A particular password may allow retrieval of all data, both general and confidential, but allow modification of nothing.
- A terminal operator may have the password to modify everything using the update network, but have no clearance to use the file inquiry facility.
- Another password may allow partial clearance for the retrieval of general data only, nothing confidential.

E. Design of On-line Updates

Simplicity and consistency should be kept uppermost in mind when designing an update network for on-line entry.

It is almost mandatory that the screen page resemble as nearly as possible the format of the document from which the operator will be keying. If it is not possible to duplicate the document format, then at the very least the data items on the screen should be in the same sequence as those on the form.

The field names of the data items on the screen should also be identical, whenever possible, to those used on the document.

It may not be feasible, and sometimes may not be desirable, to transpose the entire contents of a document page onto one screen page format. It would be better to avoid overcrowding the screen, thereby presenting a picture that would appear confusing to an operator. For most of the forms utilized by the Personnel system, three screen pages were necessary to reproduce the contents of one document. (See Fig. 5 for example of screen pages designed for use with the Personnel Action Form). It is also important to be consistent among all screen pages as to relative location of data blocks, format blocks, and message blocks.

These guidelines for simplicity and consistency not only make life easier for a terminal operator, but add a sense of structure and inherent documentation to an otherwise complex update network.

Training of Staff

One of the most important points to be made with regard to user education is that management and those responsible for file integrity be thoroughly familiar with all the data elements and with the basic organization of their data base. It is only when this is accomplished that the advantages of a data base system and on-line access can be utilized to their full potential.

In regard to training for terminal usage, it is advisable to teach people in a user office only those operations which they will have to perform on a day-to-day basis. For example, any effort to teach a terminal operator to use the query facility when his job will only entail keying for the on-line update will only serve to confuse. Don't start to train staff too early. Without immediate usage of the new knowledge, people will have forgotten and lost interest when the new system finally comes on-line. Keeping learning groups small will enable each person to have hands-on time and allow easier viewing of the keyboard.

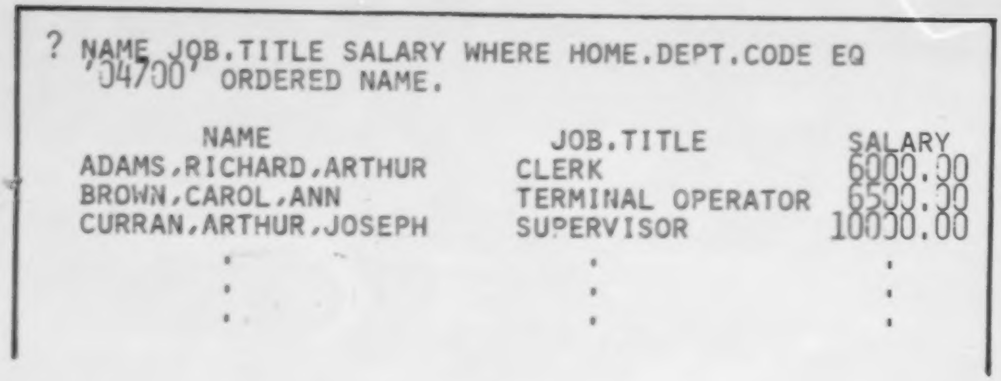
Advantages

The following are some of the tangible advantages already realized from the on-line data base Personnel System:

- File integrity has been drastically increased because of superior editing capabilities and the centralization of employee data.
- Because data entered in error is caught and corrected immediately at the keyboard, the time frame required to turn around a batch of update transactions has been sharply reduced.
- Most of those reports still requested can be submitted immediately using the batch print program or be defined in a half hour or less using the report generator.
- There is access to a higher volume of data, including five years of job status history.

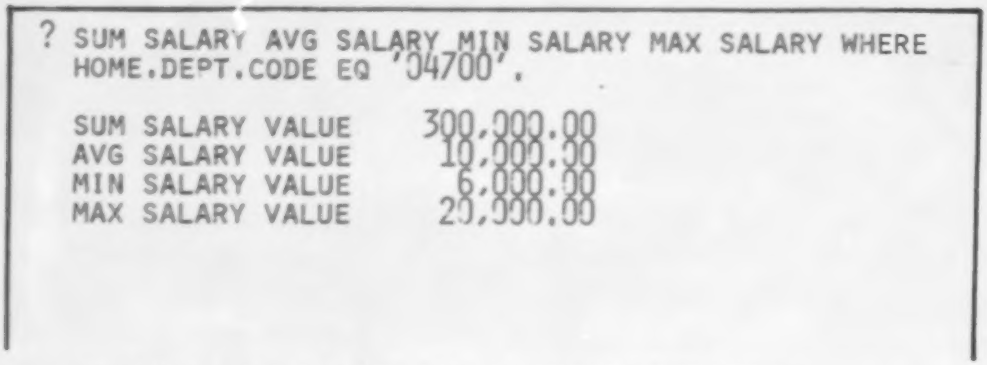
- On-line inquiry has virtually eliminated all 'one-shot' programming.
- File lookup is accomplished immediately.
- A new capability exists enabling a user to 'browse' through his file looking at data in general.
- Many questionnaires can be completed using the query facility.
- Statistical data can be computed also using the query facility.

The on-line data base for Personnel Records has resulted in a high degree of user satisfaction. But its success is due in part to the cooperation and effort expended by both the supervisory and clerical staff of the Personnel office. If the conversion to a data base system is to be successful, it must be stressed that a high degree of cooperation and willingness to learn are the tradeoffs to a system that will provide sophisticated and superior advantages. This is not to say that we were not beset by problems and subsequent reworkings, but we feel that the additional effort expended has produced a Personnel System that can be considered innovative.



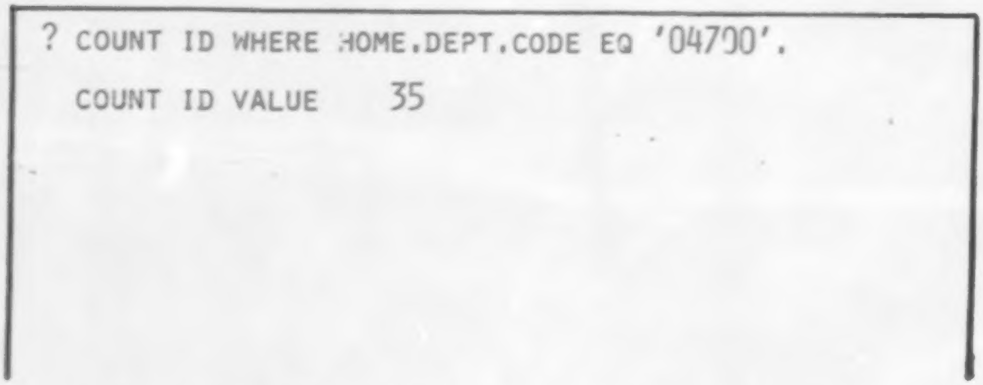
```
? NAME JOB.TITLE SALARY WHERE HOME.DEPT.CODE EQ
  '04700' ORDERED NAME.

          NAME                   JOB.TITLE          SALARY
ADAMS,RICHARD,ARTHUR       CLERK                   6000.00
BROWN,CAROL,ANN            TERMINAL OPERATOR       6500.00
CURRAN,ARTHUR,JOSEPH       SUPERVISOR             10000.00
          .                      .                    .
          .                      .                    .
          .                      .                    .
```

ALPHABETICAL LIST, BY NAME, OF ELEMENTS REQUESTED FOR A GIVEN DEPARTMENT



```
? SUM SALARY AVG SALARY MIN SALARY MAX SALARY WHERE
  HOME.DEPT.CODE EQ '04700'.

  SUM SALARY VALUE    300,000.00
  AVG SALARY VALUE     10,000.00
  MIN SALARY VALUE      6,000.00
  MAX SALARY VALUE     20,000.00
```

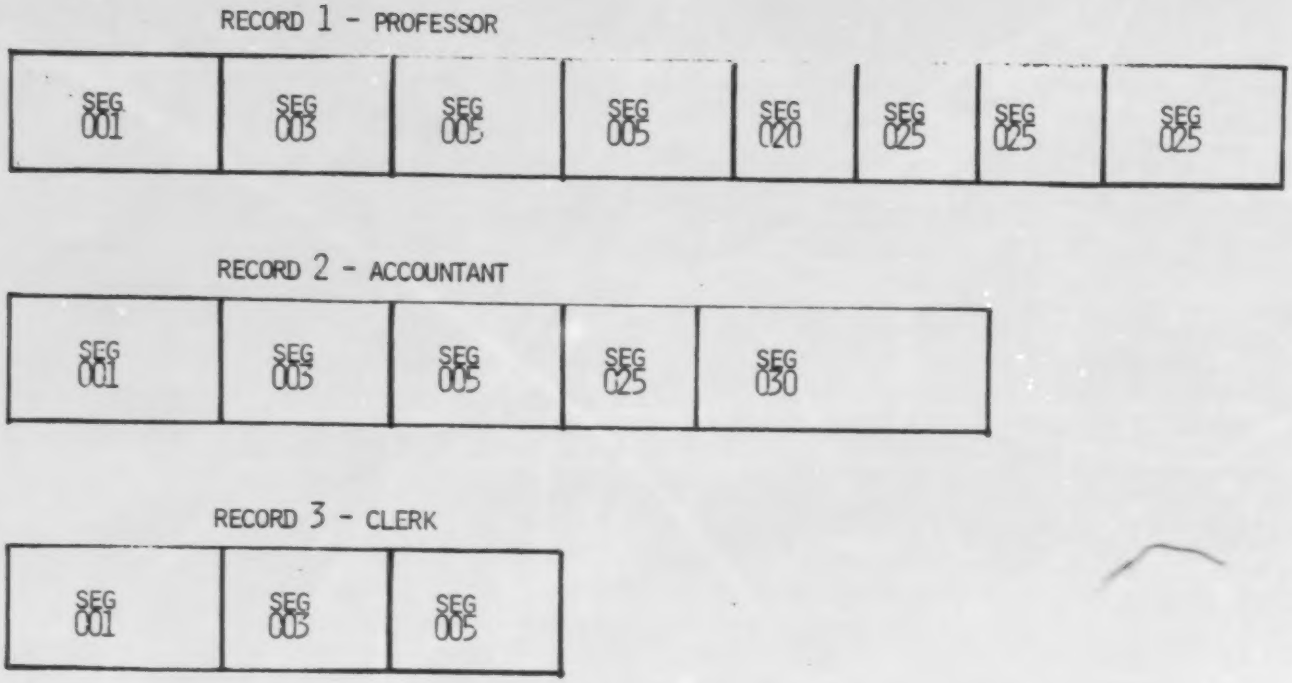
FUNCTIONS OF SALARY FOR A GIVEN DEPARTMENT



```
? COUNT ID WHERE HOME.DEPT.CODE EQ '04700'.

  COUNT ID VALUE    35
```

COUNT OF INDIVIDUALS (BY SOME UNIQUE ELEMENT SUCH AS ID) WITHIN A GIVEN DEPARTMENT

FIG. 0 - EXAMPLES OF THE GENERALIZED FILE INQUIRY SERVICE.



SEG 001 (ONE,ONE) - GENERAL EMPLOYEE DATA

SEG 003 (ONE,ONE) - WORK ADDRESS

SEG 005 (ONE,MANY) - JOB ASSIGNMENT

SEG 020 (ZERO,MANY) - FACULTY DATA

SEG 025 (ZERO,MANY) - DEGREE INFO

SEG 030 (ZERO,MANY) - PROFESSIONAL CERTIFICATION

FIG. 2 - RECORDS BROKEN INTO SEGMENT COMPOSITION SHOWING HOW DIFFERENT EMPLOYEE TYPES MAY HAVE DIFFERENT SEGMENT OCCURRENCES, THEREBY RESULTING IN OPTIMIZATION OF AVAILABLE SPACE.

UNIVERSITY OF VERMONT

DATA BASE ELEMENTS

SEG #	TITLE	DESCRIPTION	CURR (C) NEW (N)	COBOL PICT.	NEEDED FOR	POINT OF COLLECTION	UPDATING POINT

FIG. 3 - DATA ELEMENT WORKSHEET.

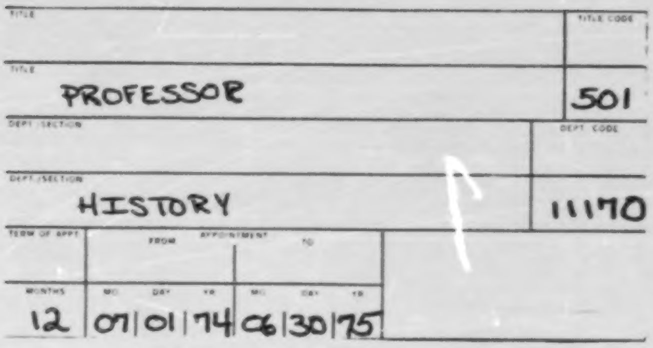

TITLE
TITLE CODE
TITLE
PROFESSOR
501
DEPT/SECTION
DEPT CODE
DEPT/SECTION
HISTORY
11170
TERM OF APPT
APPOINTMENT FROM TO
MONTHS 12
MO DAY YR 07 01 74
MO DAY YR 06 30 75

FIG. 4A - ORIGINAL FORM, ALL FIELDS ARE ENTERED.

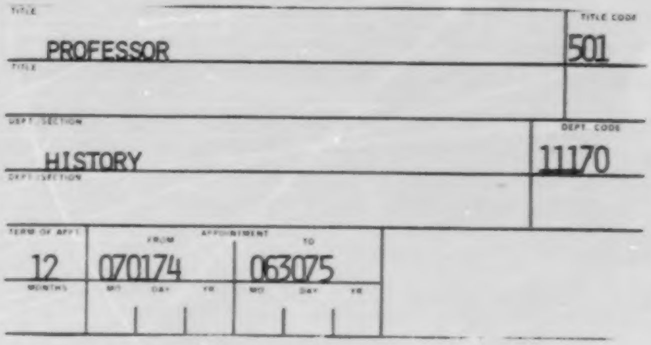

TITLE
PROFESSOR
TITLE CODE 501
TITLE
DEPT/SECTION
HISTORY
DEPT CODE 11170
DEPT/SECTION
TERM OF APPT 12
APPOINTMENT FROM 070174 TO 063075
MONTHS
MO DAY YR
MO DAY YR

FIG. 4B - BATCH REPRINT OF FORM AS ENTERED.

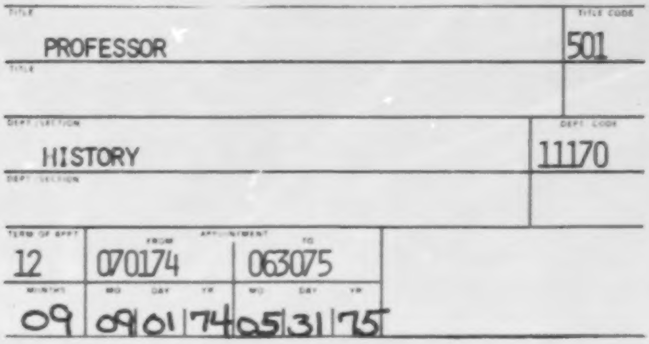

TITLE
PROFESSOR
TITLE CODE 501
TITLE
DEPT/SECTION
HISTORY
DEPT CODE 11170
DEPT/SECTION
TERM OF APPT 12
APPOINTMENT FROM 070174 TO 063075
MONTHS 09
MO DAY YR 09 01 74
MO DAY YR 05 31 75

FIG. 4C - CHANGE ENTERED ON LATEST PRINTED FORM.

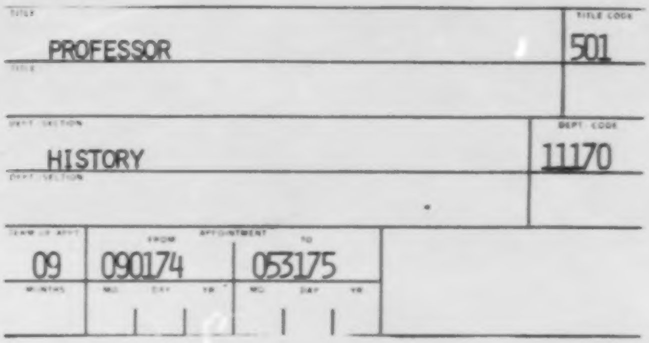
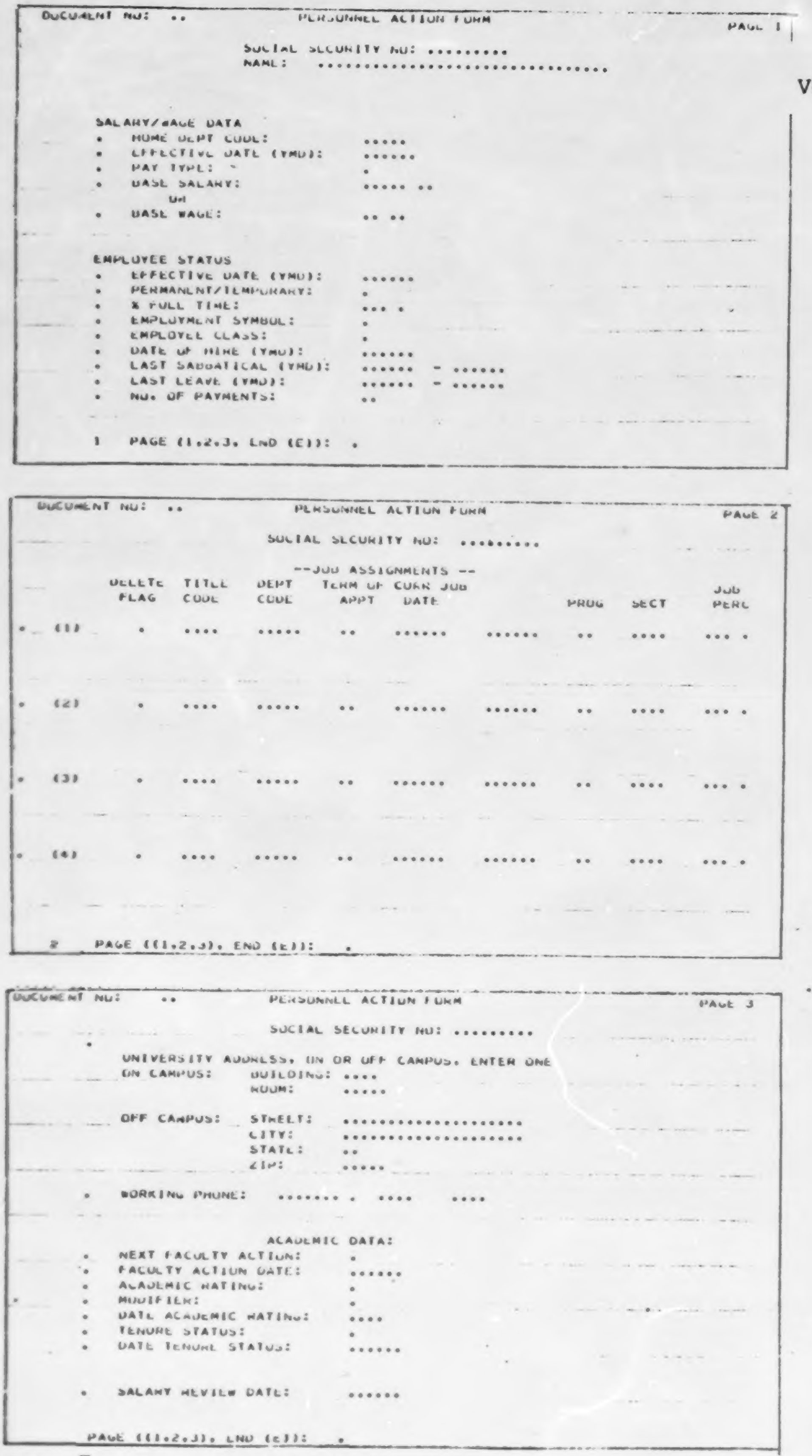

TITLE
PROFESSOR
TITLE CODE 501
TITLE
DEPT/SECTION
HISTORY
DEPT CODE 11170
DEPT/SECTION
TERM OF APPT 09
APPOINTMENT FROM 090174 TO 053175
MONTHS
MO DAY YR
MO DAY YR

FIG. 4D - REPRINT OF FORM WITH ALL CHANGES REFLECTED.

FIG. 4 - TURNAROUND DOCUMENT PROCESS. STEPS 4C AND 4D ARE REPEATED FOR ALL FUTURE CHANGES AND CORRECTIONS.



```
DOCUMENT NO:  ..              PERSONNEL ACTION FORM                          PAGE 1

                         SOCIAL SECURITY NO: .........
                         NAME:    ..............................


    SALARY/WAGE DATA
    .   HOME DEPT CODE:              .....
    .   EFFECTIVE DATE (YMD):        ......
    .   PAY TYPE:                    .
    .   BASE SALARY:                 ..... ..
          OR
    .   BASE WAGE:                   .. ..


    EMPLOYEE STATUS
    .   EFFECTIVE DATE (YMD):        ......
    .   PERMANENT/TEMPORARY:         .
    .   % FULL TIME:                 ... .
    .   EMPLOYMENT SYMBOL:           .
    .   EMPLOYEE CLASS:              .
    .   DATE OF HIRE (YMD):          ......
    .   LAST SABBATICAL (YMD):       ......  - ......
    .   LAST LEAVE (YMD):            ......  - ......
    .   NO. OF PAYMENTS:             ..


    1   PAGE (1.2.3. END (E)):  .
```

```
DOCUMENT NO:  ..              PERSONNEL ACTION FORM                          PAGE 2

                         SOCIAL SECURITY NO:  .........

                              --JOB ASSIGNMENTS --
           DELETE  TITLE   DEPT   TERM OF CURR JOB                                JOB
            FLAG   CODE    CODE    APPT    DATE                PROG   SECT       PERC

.  (1)       .     ....    .....    ..    ......    ......      ..    ....      ... .


.  (2)       .     ....    .....    ..    ......    ......      ..    ....      ... .


.  (3)       .     ....    .....    ..    ......    ......      ..    ....      ... .


.  (4)       .     ....    .....    ..    ......    ......      ..    ....      ... .


   2    PAGE ((1.2.3). END (E)):   .
```

```
DOCUMENT NO:     ..           PERSONNEL ACTION FORM                          PAGE 3

                         SOCIAL SECURITY NO: .........
      .
          UNIVERSITY ADDRESS, ON OR OFF CAMPUS, ENTER ONE
          ON CAMPUS:     BUILDING: ....
                         ROOM:     .....

          OFF CAMPUS:    STREET:   ...................
                         CITY:     ...................
                         STATE:    ..
                         ZIP:      .....

      .   WORKING PHONE:   ....... .  ....    ....

                             ACADEMIC DATA:
      .   NEXT FACULTY ACTION:        .
      .   FACULTY ACTION DATE:        ......
      .   ACADEMIC RATING:            .
      .   MODIFIER:                   .
      .   DATE ACADEMIC RATING:       ....
      .   TENURE STATUS:              .
      .   DATE TENURE STATUS:         ......


      .   SALARY REVIEW DATE:         ......


      PAGE ((1.2.3). END (E)):   .
```

FIG. 5 - SCREEN PAGES FOR ENTRY OF 'PA' FORM.

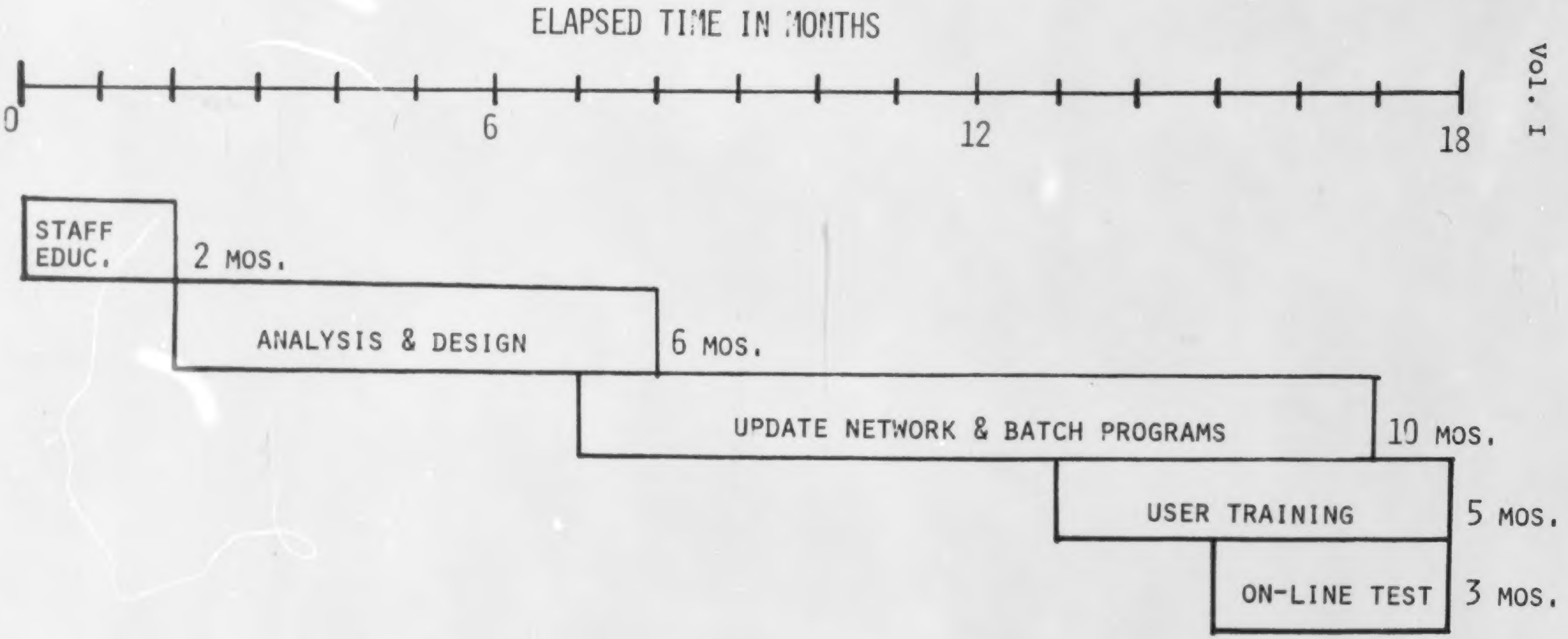


FIG. 6 - BREAKDOWN OF EIGHTEEN MONTHS OF EFFORT SPENT TO IMPLEMENT THE ON-LINE PERSONNEL DATA BASE SYSTEM. TIME FRAMES INCLUDE A LEARNING CURVE.

THE SUCCESSFUL IMPLEMENTATION OF AN AUTOMATED STUDENT LOAN ACCOUNTING AT LOUISIANA STATE UNIVERSITY

Mike Schouest
Director
Data Processing Services
Louisiana State University

J. Glenn Gremillion
Assistant Director
Data Processing Services
Louisiana State University

John R. Davidson
Computer Analyst
Data Processing Services
Louisiana State University

Leslie R. Lavergne
Director
Student Aid & Scholarships
Louisiana State University

SYSTEMS EXCHANGE CASE STUDY: THE SUCCESSFUL IMPLEMENTATION OF AN AUTOMATED STUDENT LOAN ACCOUNTING SYSTEM AT LOUISIANA STATE UNIVERSITY

HISTORICAL BACKGROUND

An account of the early attempts to automate Student Loan Accounting at LSU reads like the script of a Grade B horror movie. However, unlike the movie script where good eventually triumphs over evil, and the villian gets his just punishment, the finale' of the early attempts to automate Student Loan Accounting at LSU always ended with disaster.

The story of the automation of Student Loan Accounting at LSU can best be told with a chronological narrative. The long periods of inactivity between January 1966 and December 1972 are quite conspicuous. However, from January 1973 to December 1973, the pace quickened considerably and finally between January 1974 and July 1974, there was a virtual flurry of activity. We wish to emphasize the point that once a definite commitment was made, the automated Student Loan Accounting system was implemented in a relatively short period of time.

The first attempt at the automation of Student Loan Accounting at LSU came in 1966. Unfortunately, the project failed because the necessary data processing resources were not available.

Other stabs at automation were made during the next four years but the next serious attempt at automation did not come until 1970 when Student Aid purchased a stand-alone accounting machine for Student Loan Accounting. This project also failed, primarily because the equipment salesman grossly over sold the capabilities of his machine.

Again, a few stabs were made at automation, however the first real break-through came in February, 1973 when a systems analyst from the Accounting Services Department began an Accounts Receivable system study. This analyst was a rather ambitious fellow and quickly branched into other areas which were related to Accounts Receivable. He very quickly recognized Student Loan Accounting as a system which was very closely related to Accounts Receivable. On March 8, 1973 the analyst presented the Director of Student Aid and subsequently the Vice Chancellor for Business Affairs with a feasibility study proposing the automation of the Student Loan Accounting system. The Vice Chancellor for Business Affairs, after reviewing the feasibility study, commissioned the analyst to draft systems specifications for a Student Loan Accounting system.

On June 19, 1973, after a rather detailed study, the analyst submitted his specifications to the Director of Student Aid & Scholarships and the Vice Chancellor for Business Affairs. The Vice Chancellor, upon receiving the system specifications, directed the analyst begin a search for systems which satisfied our needs. At that time, he also directed the analyst to enlist the support of Data Processing Services in searching for these systems. The analyst was a former Data Processing employee and one with whom we were able to work quite closely, thus we began our cooperative investigation.

It was at this time that we became rather heavily involved with CAUSE. We requested abstracts on Student Loan Accounting systems from the CAUSE library and began reading papers from CAUSE meetings and from Machine Records Conferences. At this stage, the role of Data Processing Services was somewhat of an advisory nature in that all we were attempting to do was assist the analyst in obtaining various alternatives for the solution of the Student Loan Accounting problem. On September 27, 1973, the Director of Student Aid and the analyst presented several alternatives to the Vice Chancellor who promptly asked for an evaluation by Data Processing Services. These alternatives included four systems from large universities and one proposal by a commercial concern which specializes in Student Loan Accounting work. Sufficient documentation for the evaluation had been received on all of the university systems through the CAUSE library and from the universities. The Director of Student Aid was able to use this documentation in making his evaluation of their system. On October 24, 1973 Data Processing Services submitted its evaluation of the various systems to the Vice Chancellor and recommended that a particular system be acquired through CAUSE. It was our opinion, at that time, that although this system did not exactly meet our needs, it could be adapted and was a system that could be implemented in a minimum amount of time.

Some of the factors which entered into the decision to acquire this particular system were: (1) the student loan situation was very close to our own in terms of number and types of loans, (2) the type of computer and operating system was the same and (3) the system had been initially implemented at another institution and successfully exchanged twice through CAUSE. Hence, it was a system which had stood the test of implementation at three different universities and had been modified in each case to meet the particular demands of the university at which it had been implemented. Therefore, it was recommended to the Vice Chancellor for Business Affairs that this system be implemented with only minor modifications. The minor modifications would be of the nature of changing the headings of reports.

By November 30, 1973, a month had passed and still no formal response had been received from the Vice Chancellor as to his decision on the Student Loan Accounting system. It was our opinion that the new system could not be implemented without additional staff, and we had said so when the system was recommended. We felt that two additional people were needed in order to implement the system in the time that was allotted. Since the two new positions meant a continuing expenditure of funds, the University's administration was reluctant to approve the proposal. However, on November 30th, The Director of Student Aids and Scholarships and the analyst who had prepared the system specifications wrote a joint letter to the Vice Chancellor reminding him that no action had been taken on the matter. They requested a decision. On December 5, 1973, the break-through finally came! The Vice Chancellor recommended to the Chancellor of the LSU-Baton Rouge campus the approval of the project and the hiring of one high level Civil Service Analyst/Programmer. The Chancellor promptly approved the project and authorized the hiring of an EDP Specialist II as well as the allocation of funds for supplies and other miscellaneous expenses.

After a month of exhaustive searching for a person who satisfied the qualifications of the job which had been authorized, we decided to "rob Peter to pay Paul" and promote somebody from within our organization to fill the job. On January 2, 1974, an EDP Specialist I in our organization was promoted to an EDP Specialist II position and the Student Loan Accounting system was officially under way.

The first task that the Specialist performed was to spend a month in Student Aid acquainting himself with the manual system. The net result of this effort was a document which outlined all of the procedures in the manual system. After reviewing the document in some detail and after numerous discussions with the Director of Student Aid, it became evident that the system we planned was not going to satisfy the needs. Mr. Lavergne will go into more detail as to why the system was not conducive to our needs. On February 1, 1974, we came to a firm decision not to utilize the system we acquired through CAUSE. The issue was debated at length with the Director of Student Aids who agreed to wait until as late as December, 1974, for the implementation of the project if necessary rather than the original July 1 date. We decided to write our own system, and to use the conceptual approach of the systems we acquired through CAUSE.

We began by determining our system specifications. After the specifications were established, we tried to set a realistic implementation date. Fortunately, we were able to use the basic system concepts and the user documentation from the system we were initially going to implement. From telephone conversations and correspondence with the university whose system we were going to implement, we obtained information which was helpful in planning our conversion. The excellent specifications which had been drafted by the system analyst from Accounting Services were also very useful as was the material supplied by another university. Between February 1 and February 22, 1974, our computer analyst burned a great deal of "midnight oil" trying to design a basic system which met the requirements of Student Aid and one that had a schedule with which we could live. We realized that it was impossible to do all of the detail system work necessary in the time available to us, however, we believed that we could develop a basic system structure which would be flexible and upon which we could build a more detailed system. On February 22, documentation was presented to the Directors of Student Aid and Data Processing Services outlining our general systems design concepts. At that time the decision was reached that no matter what the cost, we would still try to implement the system by July 1, 1974. As you will see, our record layout contains fields which are important for fiscal year end reporting. We believed that it would be easier to implement on July 1, rather than in December because of the extra effort that would be required by both offices to update the fiscal year-to-date fields in December.

A rather detailed implementation schedule for the system and in many cases for the programs that were to be written was drafted. We had arrived at still another key decision in the process and that was to have our conversion keypunching done by a service bureau. This was suggested by one of the universities we were working with through CAUSE. The load in our own keypunch department was such that a July 1 implementation date could not be met had the keypunching been done "in house." The basic system design was presented to the two directors on February 22. John Davidson will now present our design concepts which are the basis for our existing system.

SYSTEM DESIGN AND THE IMPLEMENTATION PROCESS

After completing the design specifications and receiving approval to proceed with the system, we issued bid specifications for the keypunch work. A conversion form had already been designed and we were ready to issue bid specifications pending approval of our system design. We were confident of our decision to use a service bureau for keypunching because of the nature of our conversion form. A great deal of effort was spent designing the form since it had to be one that Student Aid could code easily and it had, at the same time, to facilitate keypunching. The layout of the form was discussed with our keypunch staff as well as with some of the established service bureaus before it was printed. The larger service bureaus in the Baton Rouge area were helpful in the design of our form, but they generally indicated that they did not care to bid on our keypunch job. The nature of the job was such that a large volume of cards had to be punched in the relatively short period of time with a very low error rate. The job required overtime work and most of the large shops felt that there wasn't enough margin of profit involved for them to submit a bid. Thus, only four bids were received. After careful review of the bids, we accepted the bid of a small vendor on March 14, 1974.

While all of the keypunch analysis was taking place, bid specifications were being drafted for our instant mailer bills. Early in March, we received two bids on the form. Even though both vendors had submitted samples that satisfied our specifications, they were so different in format and construction that we decided to seek help in evaluating the forms. We took the two samples that were supplied by the vendors to the post office on campus and discussed with them at length the construction of the forms and the placement of data on the form so that the cancellation process of the Postal Service would not obliterate any of the key data. We also discussed with the Postal Service methods of sorting the mailers prior to printing so as to expedite the processing of the mail. The postal service was extremely helpful and seemed very pleased that we had consulted them since this was going to be a large volume mailing each month. Based on their suggestions, we accepted one of the bids and ordered the mailer.

The latter part of February and the first part of March was an extremely busy time for us. As I have previously mentioned, we had already designed the conversion form and had them printed at the LSU printing office. February 18th was a "red letter" day for our Student Loan Accounting system. On that day, Student Aid began coding the conversion forms and the programming effort began on the initial file creation programs. We had determined those tasks that needed to be completed by specific dates and had outlined a very detailed schedule to meet the deadlines. We adhered very closely to that schedule even though it meant some overtime work. From the first of March until the end of April, it seemed that we were making very little progress, but at that time detail analysis work was still being performed in the Office of Student Aid and the programming effort was moving forward at a steady pace. However, no major check points occurred from March 1st until April 22nd, 1974.

The original bid specification from the keypunch work dictated that Student Aid would have all conversion forms completely coded by April 29th and that the keypunch vendor would have all forms keypunched and verified within a 10 day period after receipt of the documents. Since Student Aid had a number of the forms already coded by April 22nd, the keypunch vendor was contacted to determine if he wished to start punching early. Of course, he was quite happy to do so. On April 22nd, keypunching began on those forms that had been completed by Student Aid. By April 23rd, the vendor had completed the first batch which was then processed through our edit and balancing program. On April 29th, according to schedule, Student Aid completed the coding of all conversion forms and on May 8th, some two days ahead of schedule, the vendor completed keypunching all of the forms.

We had realized early in April, however, that we had made one serious error in our design and that changes to the system were necessary. It was our intention to distribute payments utilizing a computer program. However, we realized when we started distributing payments in the parallel runs that although our calculations were essentially the same as those performed manually by Student Aid, there were minor differences which caused discrepancies in our totals. In other cases, the computer calculations were substantially different from the manual ones because Student Aid was performing some calculations which were applicable only to very special situations. We realized that the Student Loan Accounting Master file could not balance to the manual file at the end of June if we calculated with the computer for our file and Student Aid calculated manually for the ledger cards. We also realized that it was imperative that the two files balance to the same General Ledger controls on June 30th. We decided to have Student Aid manually split payments between principal and interest through the 30th of June for our computer parallel runs. We continued to run the program which automatically distributed the payments and compared that which Student Aid distributed manually to that which had been calculated by the computer program. Discrepancies that were detected were very carefully analyzed by Student Aid and in those cases where the program was wrong, we corrected it. We came to the conclusion that we could calculate the vast majority, perhaps some 95% of the payment distributions initially and probably an even greater percentage in the future. But, we also realized that there were some situations in which we could never distribute a payment, and that we needed to give Student Aid the flexibility of distributing some payments manually. That is how the system is working today. The vast majority of the payments are distributed by computer program, but Student Aid does have the flexibility to distribute manually.

On May 10, the initial Student Loan Accounting file was created and due to an excellent conversion effort, the run balanced to General Ledger controls of over six million dollars the very first time that the file was created. On May 13th, we began processing payments and transactions as a parallel operation for the period of April 16th through May 15th. This constituted the first month of the automated Student Loan Accounting system. At the same time, Student Aid continued to process payments manually as though the new system did not exist. We processed data for the period May 16th through June 15th in the same manner with a target date of June 24th for printing the first bills for the period ending June 15th. We did not plan to print bills for the period ending May 15th We planned to continue the manual system through June 30th and at which time the ledger cards would be closed and the new fiscal year would begin with the automated system.

By June 6, 1974, all of the data for the period of April 16th through May 15th had been processed. The Student Loan Accoutning Master file was updated through that period. By June 18th, all the data for the period of May 16th through June 15th was balanced and the Student Loan Accounting Master file was updated. Finally, on June 24th, 1974, a little more than six months after we began work on the system, the first Student Loan Accounting instant mailers were printed and mailed.

The remainder of June and part of July were dedicated to the close-out of the fiscal year. On July 23rd, the final June 30th close-out was made. At that time the Student Loan Accounting Master file was updated with transactions and daily payments which had been received through June 30th, and the fiscal year-to-date fields were zeroed. This process initialized the file for the beginning of the new fiscal year. All the data to make runs for the month of July was available and the next day the July bills were printed and mailed.

At that time we expected to be basically operational, however, in the month of August a large number of advances were to be recorded in the Student Loan Accounting Master file. Fortunately, the August bills were run on schedule, by August 24th.

The months of September, October, and November had proven to be typical months in that operations had been able to run the system with very little analyst intervention with an average of 20 hours of computer time each month. Although the hurdle of the year end programs has not been crossed, we do have an operational monthly system which not only satisfies our users' requirements but also runs smoothly in our operations environment.

We have proved that in a relatively short amount of time, basically with the resources that are in house and with the information obtained from other universities, a Student Loan Accoutning System can be developed from its inception to the completion of the monthly cycle in something slightly more than six months. We think that any university can accomplish this task with proper planning and support from its management, as well as its users.

A SYSTEM DESCRIPTION

The general thrust of this part of our presentation is to acquaint you with our system to the extent that you can fully appreciate our design and conversion problems.

Before we begin talking about our automated Student Loan Accounting System let me give you a feel for our computer environment. Exhibit 1, which is self explanatory, indicates both the configuration of our system as well as the necessary requirements of the Student Loan Accounting System.

The cornerstone of any system is its master record. The Student Loan Accounting Master record is a variable length record containing one 200 byte section for each borrower and up to five 150 byte sections for each loan. The fixed portion of the record is divided into control fields, demographic data, Student Aid data, and a few other items that we have found useful in other master records. One point worth mentioning here is that the record key for our records is not the social security number but a borrower number. Social security number is considered to be demographic data.

Each variable portion contains information about a different loan. Currently only NDEA and NDSL loan are processed by our system, and we consider NDEA and NDSL loans to be different even though a student had one loan which was continued from NDEA to NDSL because of legislative action. If a student with a previous loan returns to school and receives another loan, we consider this a new loan.

Before we cover the system flow charts, it will be worthwhile to stop a minute and look at the bills that we are sending to the borrowers. This form is an instant mailer which is constructed in such a way that if the borrower uses the return envelope provided to return his check, we will have a copy of the bill which he is paying.

The form provides for separate listing of the note balance of each loan and the amount due for each loan. Payment amount and the account number are displayed twice to allow for possible obliteration in the postal cancellation process.

with the amount printed and if they are different, the actual amount of the check is recorded above the printed amount. On a daily basis these bills are sent to Data Processing Services for keypunching (Borrower number and amount) and processed in order to give Student Aid a daily alphabetical list of payments.

Our flow charts are organized around control points in the system; i.e. points at which controls must be balanced or edit errors must be corrected. In the Daily Payment Batches process, the account number and the amount are edited and an error list is printed. When all errors have been corrected, a list in alphabetical order of the day's payments is produced. This list then serves as back up for deposits.

Our maintenance system is not unusual, however, the following features are significant: (1) we provide for recycling of maintenance cards on disk. This means we do not have to manually process correction cards, (2) we use maintenance cards with change codes, (3) we have a combined edit-maintenance, (4) we do not update in place. We update a sequential tape with a disk output. Consequently, the edit maintenance program may be run as many times as necessary, (5) we do not have, nor do we need, a method of changing a borrower number. There does exist provisions for updating the social security number, (6) maintenance does not provide the capabilities for updating the principle balance field, (only a transaction or a payment may update this field.), (7) borrower numbers are assigned by the maintenance program, not by people.

The third process is the editing and balancing of the miscellaneous transactions. Advances, return of checks, cancellation of advances, etc. are all considered to be miscellaneous transactions. The monthly miscellaneous transaction cards are edited against the output Student Loan Accounting Master file from the current month's maintenance process. Diagnostics and a proof listing with balancing totals are produced as a result of this process.

The fourth process is the transaction and billing update portion of our system. Daily payments and miscellaneous transactions are used to update the disk Student Loan Accounting Master file which was output from the maintenance process. These transactions are captured for historical purposes on a yearly accumulation basis. We do, of course, print a transaction register indicating the transactions for that month as well as a list of the loans that have been completely paid out.

Next comes our third and final update program of the system, the monthly billing update. This program determines what is to be included on the statement. It also accounts for the number of months that a person is behind in his payments. The printing of the instant mailer bills is delayed until all runs are free of diagnostics and in balance. One of the most important runs to be checked is the trial balance list which is printed from the final output Student Loan Accounting Master file by the billing update program. Also from this same final Student Loan Accounting Master file, Work Papers are produced indicating such things as number of people delinquent, (and how long) and people whose status end date or status review dates have arrived. A billing register is printed indicating the information that we printed on each bill. Note that the bills are printed from a Billing File not from the Student Loan Accounting Master.

There are some housekeeping chores of our system. Each month we backup our final Student Loan Accounting Master file to another tape in case we cannot read the original one. Also at the end of each month, we reload our Student Loan Accounting Master file to disk for use in editing and listing the daily payments.

A subsystem prints record dumps for every person who had had record maintenance during the month. The latest of these record dumps is placed in the borrowers folder in Student Aid, and the one which is already there is removed. This gives Student Aid a record of the contents of a borrowers record without having to contact Data Processing Services. So far, we have discussed the daily payment system and the monthly system. It is important to note that the monthly system is run 13 times a year. Normally our cycle runs from the 16th of one month through the 15th of the next month. However, we also have a June 30th close out each year for which we process the equivalent of a monthly run but only for the period June 16th through June 30th.

The monthly billing update program is not run at fiscal year end. At fiscal year end, we bypass the billing step and produce a fiscal year-end tape and a backup tape. We drop loans which had been payed out and zero fiscal year-to-date fields creating a new fiscal year beginning Student Loan Accounting Master file and a disk file indicating the records we have dropped as well as those that were retained. Transaction totals for that fiscal year and similar information from preceeding years as well as update cards for correction purposes gives us the data needed for the annual federal reports. We do however, get some information directly from our final fiscal year end master file.

All systems described, with the exception of the fiscal year-end subsystem, are operational. We currently have provisions for printing Student Loan Accounting Master file record dumps from any master file in the system. We also have the capability of printing transactions from any year-to-date accumulated transaction file.

THE SYSTEM AS VIEWED BY THE DIRECTOR OF STUDENT AID

At the present time our student loan accounting system is used only for National Defense/Direct Student Loans. This is how it works:

1. An account is set up when a loan is committed to a student. A load form listing the loan amount and certain demographic information is then coded by our department and sent up to keypunching. A borrower number is assigned by the computer.

2. When checks are to be written for loans, we request a printout of all commitments made. This comes to us alphabetically and can be used as our check request voucher to the Accounting department. (Supplemental checks for loans made after this are handled as individual transactions.)

3. When the borrower leaves school, a repayment schedule is set up for him by our office and then coded for keypunching with interest start date, payment due date, amount due, etc.

4. Ninety days before the borrower's first payment becomes due, Data Processing will send us a label with his name and address preprinted. At this time we will send the borrower a form letter noting that his first payment is coming due. With this we will include a pamphlet telling

him how interest will be computed, how to read the billing statement deferment and cancellation privileges, and giving him other pertinent information which may head off future questions and collection problems down the line.

5. All payments become due as of the 5th of the month and all loans made since 1968 are due on a monthly basis. Billing statements are run about the 22nd of each month, sorted by zip code number, and delivered to the post office. (Data Processing sends my office a register showing all borrowers being billed.) The statements are addressed by computer on forms already sealed with return envelopes enclosed. No individual stamping is necessary because we use a first class postage-paid permit. Then the total cost of each mailing is charged back to our department.

6. When the borrower receives his bill and opens it, he will find a statement showing the amount of principal and the amount of interest currently due and past due on each type of loan that he has as well as the total outstanding balance of each of his loans. In bold print, he will see the amount he is being asked to pay. A message will also be printed on his statement requesting him to pay by the 5th of the month and to send his payment in the return self-addressed envelope enclosed. Now, if he does this, everything will go well because the interest due amount on his statement will not change until the 16th of the month. Furthermore, if he uses the return envelope--as most will--we automatically have a copy of his statement preprinted on the inside of the return envelope. It was our thinking that this would produce better results than asking him to return a statement copy. Also we attempt to capture address changes on the self-addressed envelope by asking the borrower to note his return address if this is a new address.

7. When payments are received in my office the following procedures take place: The amount of each check received is compared to the amount due on the statement. If there is a difference, a line is drawn through the amount on the statement and the actual amount received is then written in. Payments received from borrowers who choose not to use our return envelope are typed on a list giving borrower number, type of loan, and amount received. No time is spent in attempting to alphabetize the statements.

An adding machine tape is run on all checks received; then a companion tape is run on the amounts shown on the individual statements as well as on the payments listed for borrowers for whom we have no return statements. The two tapes must agree before anything can move further.

When the tapes agree, a deposit ticket is prepared and then everything is microfilmed--statements, lists, checks, tapes, and deposit slip. The deposit is then delivered to the Bursar's office while the statements and list are sent to be keypunched with an accompanying batch control sheet. Within 24 hours we will receive from Data Processing a totaled alphabetical listing of all payments received for a particular day's deposit. The total here must agree with our deposit total. Data Processing also returns to us all documents from which keypunching is done, so when we are out of balance we first check for the possibility of a keypunch error or an ommission. If that does not resolve things, then we review our microfilm tape for that day's work.

8. All maintenance is processed about the 15th of each month and a trial balance is run on all accounts. Two runs are made--one alphabetically and one by borrower number. Totals are reconciled with the General Ledger. To aid in this, we will also receive a complete listing of all transactions during the month such as the distribution of principal and interst to borrowers' accounts for payments received, correcting entries, etc.

9. After this, the procedure starts all over. But the message on the borrower's statement might be different from that of the previous month. If we didn't receive payment from him we will send him a friendly "dunning" message. If he misses two payments consecutively, the message changes to slightly unfriendly. This message is then repeated once more before we take the account out of the billing cycle for more direct attention. At this stage we will then send him a very unfriendly personal letter informing him that his account will be referred to our attorneys unless he makes proper arrangements within a specified period of time.

Another message that we send borrowers, which might be noted, is one which accompanies our January statements. On this statement the borrower is advised of the total interest he has paid during the previous calendar year.

10. A report showing the aging of delinquent accounts is to be prepared monthly, but this is still in the programming stage.

11. At the end of any given semester our student loan file is compared with the master tape file of students enrolled and we are provided with a listing of our borrowers who are no longer enrolled.

12. At the end of the fiscal year, we are provided with any statistics we request for the purpose of preparing reports. These statistics are gleaned from the demographic information coded in setting up each loan.

What about teacher cancellations? Well, after much deliberation we agreed with Data Processing that we should continue to manually calculate the principal and interest distribution on these. There were just too many variables to make it worthwhile to program automatic distribution of principal and interest. We would likely spend as much time assembling and coding the information necessary to do this as we would in making the calculation right off (not to mention the time spent in writing programs). Anyway, with the 1972 changes in the law regarding cancellations, the number in effect should decrease considerably in future years.

What about General Ledger accounting? We did consider the possibility of having the system provide automatic entries to the General Ledger but we backed away from that--at least for the time being. It doesn't take much effort or time for our office to prepare the summary entries necessary, and in doing this we can maintain direct control. This, we believe, gives us a bit better "feel" for solving reconciliation problems.

We are very happy with the way interest computations are managed. The computer does it all, except for cancellations and for certain accounts which require special handling.

It should be mentioned, too, that my staff is pleased with the fact that there are only four different forms that they must learn to code--one to set up a loan, one to signify a status change, (including address changes) one to record an accounting transaction that cannot be done by a mass listing, and one to set up the borrower's repayment terms.

Without a doubt, we are an extremely satisfied user. Yet, there have been some problems--like with the borrower who received a bill for a payment due of some $6,000 when he had borrowed a total of only $2,000. (It seems that in coding his interest start date, instead of the year 1970, we put 1907.) But we believe the problems since this was begun have been minimal. We believe that our Data Processing people have done an exceptional job of organizing and programming the project, providing excellent instructions for coding, and keeping it all tied down to a very finely tuned time schedule. We are forever grateful for the service they have rendered our department and the University.

THE SYSTEM AS VIEWED BY THE ANALYST

Some very definite conclusions had been reached as to just exactly what we wanted to do--thanks to the help provided by the systems analyst from Accounting. We believed that any system activated would have to be developed along the following lines:

1. In the receiving of repayments from borrowers, after opening the mail and preparing the deposit slip, everything necessary to update the borrower's account should be done automatically. Again, no matter how many checks are received for any one day. The system should enable us to have that deposit ready on that day and enough disc tracks left so that all accounting records could be reconciled with that deposit within a 24 hour period. No longer should a large deposit take us several days to put together and no longer should we have to handle a payment received five different times.

2. Interest charges should be done by computer, not in The Student Aid office, and distribution of principal and interest borrower's account should be automatic.

3. The system must account for new loans as well as those in the repayment stage and it must include the gathering of demographic information about each borrower so that statistical reports could be speedily prepared.

4. It must be able to account separately for loans made before July 1, 1972 under the old NDEA regulations so that we could apply the proper cancellation privileges.

5. It must be able to provide automatic aging of accounts for management review.

6. It must provide an alphabetical listing of all borrowers, the amounts they had borrowed, and the outstanding balances.

These items, we believe, were absolute necessities. Anything else could be negotiated. But whatever we did must be done fast. We simply did not have enough staff to keep things under control much longer. What was more, we knew that for a certain period of time we would have to operate parallel systems. Therefore, it was of paramount importance that we come up with a tight schedule for programming and conversion that we would all agree to meet no matter what. Without this the whole project would blow apart.

In the meanwhile, The Accounting and Data Processing departments were researching systems in use at other shcools. One of the universities was quite willing to provide documentation and source computer programs through CAUSE, any costs involved would be in-house. This approach made sense to the University administration, so this is the direction we took.

In my office, we felt we were well prepared for the task before us. Six months before we had hired a very bright young lady to take charge of our collections. She had been with us as a student worker for four years and knew the entire operation as well as anyone. More importantly she had the knack of being able to grab hold of a new concept, see the purpose behind it right off, and carry it through to fruition. In turn, she had highly competent staff members and student employees working with her who knew the vital importance of recording information correctly the first time. Although they knew little about automation, we felt sure that in the conversion process, they would do an excellent job of coding loan sheets. We had the right people to do the job; we knew what we wanted to do; we were ready to go.

Then began a long series of meetings. After about a month of talking and research, we were ready to start putting things together. And then we had a problem--a problem of such proportions that either we or Data Processing would have to change perspectives in the type of student loan accounting system that LSU was going to institute. The problem was that in the system we had acquired specifications called for a manual calculation of interest everytime the payment sent by the borrower varied from the preprinted amount on the statement. Furthermore, the specifications called for all loans under any one loan fund to be combined together for distribution of principal and interest. So, if we received a payment from a student who had both a loan under the old NDEA regulations and one under the new regulations, we would have to find a way to manually split the payment and calculate the interest to be prorated to each loan--if we wanted to retain separate accountings for each type. Well, we envisioned having at least 2,000 borrowers who had both types of loans as well as numerous others who would send in a different amount from that on the statement. Long ago in the past I had come to the conclusion that it was a gigantic waste of time for us to be computing interest in my office on adding machines and small calculators. Certainly, our computer in Data Processing could do this. If, indeed, we would be back to manually calculating interest on about 50 percent of the monthly repayments we received, I wondered if we were really gaining a great deal.

Hurriedly, we put our heads together. A possible alternative to solving the problem for persons with two different types of loans was to send them individual statements for each and request separate checks when paying each. This seemed very awkward because to the borrower it was all one student loan made at LSU. He was quite likely to put it all together and send one check, anyway; and once he did this, we were back to manually splitting the distribution of principal and interest. I revolted. Data Processing saw my point, but to do what we wanted was such a drastic departure from the system that we were adapting that, in effect, what we were requesting was a tailor-made LSU system. This would cost more initially both in time and in money, and it would alter considerably the date by which we had anticipated having things going. Nevertheless, the benefits reaped over the long run in doing this the way we wanted did seem to far overshadow the problem of the mounting workload caused by any delay. We were convinced, so was Data Processing, and so was the University administration.

The evoluntionary process was now complete. The University was committed to a tailor-made student loan accounting system.

AN OVERVIEW OF INTERPRETIVE STRUCTURAL MODELING

Dr. David W. Malone
Principal Systems Planner
Systems Planning & Analysis
Battelle Memorial Institute

An Overview of Interpretive Structural Modeling

"Interpretive Structural Modeling" (ISM) is a rather ominous looking label for a fairly simple process which is intended to help people think and communicate more effectively about complex issues. The technique has been implemented in such a way that people are responsible for making subjective judgments, while a computer is used in an unobtrusive manner for bookkeeping and for performing and displaying the results of simple logical operations.* Preliminary versions of the computer programs were available in the Fall of 1973, when the first computer-aided experiments in the [illegible] of the technique were begun.[4,5]

The fundamental concepts of the ISM process are an "element set" and a "relation". The element set is a list of items (nouns or noun phrases) which are considered to be important in some situational context, say: the members of the management staff of a corporation. The relation is a transitive verb or verb phrase expressing a possible mode of interaction among the elements, for example: "reports to". The interpretive structural model corresponding to this example element set and relation would be the organization chart of a corporation. The ISM process is concerned with the systematic development of structural (or network) models in complex situations in which the character of the elements and the connectivity are not as straightforward as in the case of the organization chart.

More generally, the elements correspond to the nodes on a network model, and the presence and direction of the relation between any two elements is indicated by an arrow connecting those two elements. For those familiar with the relevant mathematics, there is an equivalent binary matrix representation of a network, in which the elements form the contents of the index set for the rows and columns of the matrix, and the presence of the relation directed from element i to element j is indicated by placing a 1 in the corresponding intersection of row i and column j. It is important to note, however, that it is not necessary to understand the underlying mathematics to use the ISM technique, since all communication with the computer is carried out in plain English. In conducting an exercise with a group, for example, the computer is programmed to present the participants with a sequence of questions of the form "Is element i related to element j?", or a similar form more appropriate to the grammar of the elements and relational statement. The consensus view of the group as to whether the answer to this question is "yes" or "no" is then entered into the computer. (This determines whether a 1 or a 0, respectively, is entered in the appropriate location of the binary matrix.) As responses to these questions are accumulated in the computer it is possible to infer the answers to some questions based upon previous answers. The computer is programmed to make such inferences, so that the participants in the exercise need not directly respond to every possible pair-wise combination of elements.*

---

* The mathematical foundations of the technique can be found in various reference works, e.g., Reference 1. The philosophical basis for the development of the ISM approach has been presented in Reference 2, and the conceptual and analytical details of the ISM process are outlined in Reference 3.

Figure 1 is a representation of the principal operations involved in applying the technique in partnership with a computer. People are assumed to draw upon their own knowledge and attitudes to identify pertinent concepts and relationships--these "mental models" are formalized here as element sets and relations. The embedding operation is performed jointly by man and machine. The computer is supplied with an appropriate list of elements and the definition of a pertinent relation. A systematic sequence of queries is then generated, and a binary matrix representation of the system is assembled from the responses provided by a person or group of persons. (Those readers who are not interested in what goes on inside the computer may skip the rest of this paragraph.) When the matrix model is completed, computer operations are performed in order to partition the elements into natural hierarchical levels (partitioning) and to establish a minimal set of linkages which capture the entire pattern of the relation (extracting). The multi-level directed graph (digraph) which results can be inspected and interpre-

---

* This inference capability is based upon the assumption that the relational statement, which was assumed to be "transitive" in a grammatical sense above, is also "transitive" in a mathematical sense. That is, if element i has been determined to be related to element j, and element j has been determined to be related to element k, then mathematical transitivity provides that, by implication (inference), element i is related to element k. Some transitive verbs may not be mathematically transitive, e.g., "loves".

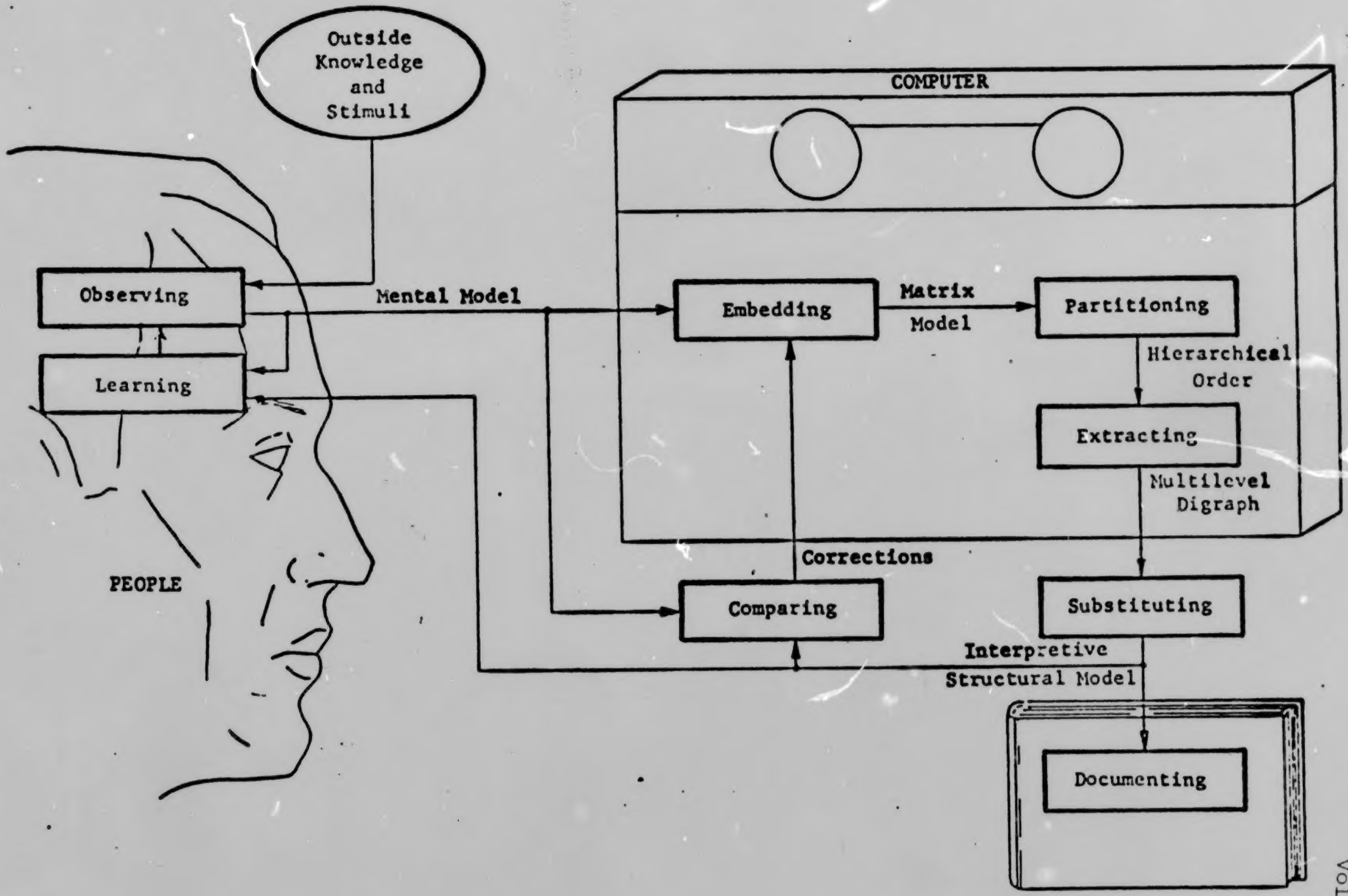


FIGURE 1. FUNCTIONAL REPRESENTATION OF INTERPRETIVE STRUCTURAL MODELING, SHOWING POTENTIAL FOR MAN/MACHINE SYMBIOSIS

tive symbols introduced according to the context (substituting) to produce an interpretive structural model. If the creators of the model are satisfied, it can be documented for purposes of communicating it to others. If there are deficiencies or errors in the model (comparing), a correction process can be executed, again in partnership with the computer. Note that the entire process has learning implications for the participants, as well as providing documentation of the results of the exercise.

The principal advantages of the ISM process are the following: first, no knowledge of the underlying mathematics of the process is required of the user, who need not even be concerned with what a "matrix" is; he simply must possess enough knowledge of the context of the exercise to be able to respond yes or no to the sequence of queries. Second, the process is systematic; the computer is programmed to consider all possible pairwise interactions of the elements, either through responses of the users to direct queries or by inference based upon responses to previous queries. The third advantage of the methodology, then, is the efficiency gained from this use of inference, which may reduce the number of queries to from twenty to fifty percent of the total number possible.

To summarize, there are five basic concepts and three basic operational steps which users of the ISM process must be familiar with. These basic concepts and steps are shown diagrammatically in Figures 2 and 3. The first operational step is concerned with extracting a set of relevant elements and a meaningful relational statement from a specified issue context. On the basis of a systematic investigation of whether or not the relational statement holds among pairs of elements, and using computer aids to perform routine bookkeeping and logical operations, a complete relational pattern is established during the second operational step. This relational pattern is established in the form of a "directed graph", or digraph. In the third operational step, the digraph is examined for completeness and possible reinterpretation of relational links, perhaps even iterating through the computer-aided step with revised element sets or relational statements. Appropriate interpretive symbols and/or notation are then introduced to produce an interpretive structural model.

A list of illustrative contexts in which the ISM process has been used to date is shown in Table I.

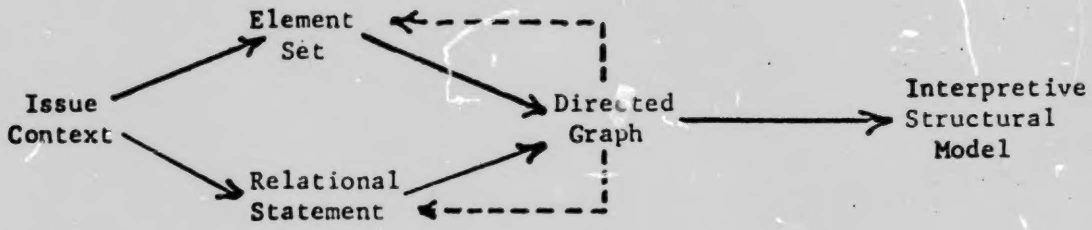


FIGURE 2. THE BASIC CONCEPTUAL ELEMENTS OF THE INTERPRETIVE STRUCTURAL MODELING TECHNIQUE (Arrows denote the presence of activities whereby some elements are examined and elaborated in order to determine others; the dotted arrows denote a feedback, or evaluative, activity.)

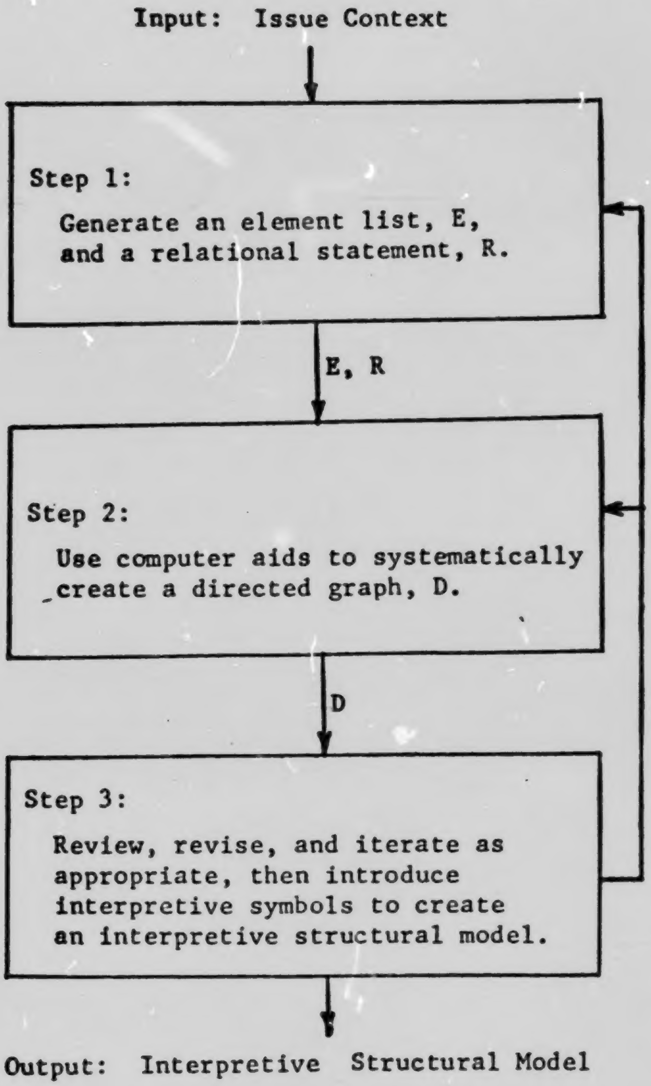

Input: Issue Context
Step 1:
Generate an element list, E, and a relational statement, R.
E, R
Step 2:
Use computer aids to systematically create a directed graph, D.
D
Step 3:
Review, revise, and iterate as appropriate, then introduce interpretive symbols to create an interpretive structural model.
Output: Interpretive Structural Model

FIGURE 3. THE BASIC OPERATIONAL STEPS FOR APPLICATION OF THE ISM TECHNIQUE

TABLE I

ILLUSTRATIVE APPLICATIONS OF ISM

- Factors in Inner City Investment Decisions
- Grouping of Children With Learning Disabilities
- Factors in Technology Assessment
- Solving Complex Econometric Models
- Structuring the "Goals for Dallas"
- Goals for National Research Council Board on Engineering Manpower and Education Policy
- Goals for a State Agency
- Citizen Participation in Studying Urban Crime

REFERENCES

1. Harary, F., R.Z. Norman, and D. Cartwright, Structural Models: An Introduction to the Theory of Directed Graphs, John Wiley and Sons, 1965.

2. Warfield, J.N., "An Assault on Complexity", Battelle Monograph No. 3, Battelle Memorial Institute, Columbus, Ohio, April 1973.

3. Warfield, J.N., "Structuring Complex Systems", Battelle Monograph No. 4, Battelle Memorial Institute, Columbus, Ohio, April 1974.

4. Christakis, A.N., and Malone, D.W., "A Systematic Approach to Human Settlement Planning", The Academy for Contemporary Problems, Columbus, Ohio, January 1974.

5. Malone, D.W., "An Introduction to the Application of Interpretive Structural Modeling", to appear in IEEE Special Transactions on Social Systems Engineering, early 1975.

TELECOMMUNICATIONS SYSTEM CASE STUDY: STUDENT REGISTRATION AND SCHEDULING

(CROSSING OVER: BATCH TO ON-LINE)

Joseph A. Duerstock
Director
Systems Planning
Miami-Dade Community College

Crossing Over - Batch to On-line

Joseph A. Duerstock
Director Systems Planning
Miami-Dade Community College

A Comprehensive Student Records Project

The Changing Environment

Miami-Dade Community College is a large (45,000 student) urban commuting college with three major existing campuses, another approved as a medical center campus, various smaller centers, and a commitment to serve even smaller groups under the umbrella of our outreach programs. This diversity of programs and locations has been emerging over the past five to eight years and has been very difficult to handle both from an administrative and a data processing point of view. The systems and procedures that worked well for a single campus or even multiple campus traditional, semester oriented college would not work in this changing environment. Classes were starting and ending at times that did not fit into the semester calendar, some finishing before the end of the semester while others continued into the following semester. Special "mini" semesters were started to help bridge the time between the quinmester schedule of the high schools and our regular semester. The long and short of it all was that the regular credit course offerings were being scheduled much like its step-sister, continuing education.

Clearly, two opposing needs emerged during this time: Classes needed to be scheduled without regard to semester boundaries and the faculty wanted to be able to modify the standard teaching modes. Although the traditional classes would continue to be the dominant form of instruction, the status quo had to give way to new techniques.

Time to Make a Change

The need to make changes in the traditional registration system in combination with the recognition by the college administration that class time patterns and hence manual scheduling would become more complex as these new teaching modes expanded led to the authorization to implement a batch oriented class scheduling system. At first this system was perceived to be the only major effort under the then defined scope of the project. But, as the committee workers developed the system objectives and specifications, it became obvious that a lot of time and effort would be wasted if we were to return to the arena type registration after the initial batch of students were scheduled by the system.

The second systems effort then developed in order to recover the seats from the drop/add processing necessary to clean up the problems after the schedular was run. It was agreed that the problems could be handled if drop/add and walk-on registrations could be processed by an on-line system.

The Scope Mushrooms

That additional system requirement, although not sounding too involved as compared to a complex scheduling program, started a series of events that, like dominos, fell one upon the other to expand the system effort to ten-fold its original size.

The domino theory proceeded as follows:

1. The on-line system requirement forced a change from sequential computer files to random access.
2. This change gave us the opportunity to add certain badly needed new data elements to the system which had not been done because of existing file structures, to eliminate most of the redundant data, to combine many files and to restructure the file organization to carry data from multiple semesters.
3. The increased availability of elements in the data base led to additional report requests although bridge programs were written to transfer and reformat data from the new system to the old.
4. The need to keep an accurate count of the number of seats in each class section dictated that the class schedule be on-line.
5. The need to have the class schedule on-line meant that the Course Catalog had to be on-line.
6. The need to process students during the days of walk-on registration meant that Admissions also needed to be on-line.
7. Since both the Admissions and Registrar needed to be able to change certain data items immediately, a records maintenance system was also needed on-line.

Drawing New System Boundaries

The scope of the project had now been redefined to include plans which would ultimately lead to the complete replacement of the existing registration system. This new system would be able to handle the known changes in the operational environment of the college.

The college administration placed paramount emphasis on these goals:

1. To relieve the student from the burden of manually constructing a schedule especially since course time patterns are going to become more complex.

2. To build a master schedule based upon student demand.
3. To gain better control of the average class size.
4. To reduce the money needed to support part time and overload teaching.

For the department chairmen who participate in the building of the master schedule, these goals were set:

1. To provide information concerning the courses that students wanted to take and the times that they wanted them.
2. To allow the student to request a course even if it were not planned to be offered.
3. To provide course conflict reports in order to reduce the number of course conflicts that are built into the class schedule.
4. To allow changes to be made to the schedule until the last moment without making students reshuffle their schedule.
5. To allow courses that are taught using different modes of instruction to be identified using a course suffix.

The goals of the Registrars tend to be a mixture of both operational and managerial.

1. To allow for flexible terms and variable credit.
2. To facilitate multi-campus registrations.
3. To eliminate all computer cards.
4. To recover seats for dropped classes.
5. To make the drop/add process easier.

The decision to undertake this massive system's effort thus envolved not so much as the result of a single user but more from a combination of those involved in the day-to-day operation of the college along with the recognition by the college administration that the time had come for change. All in all, this was the perfect environment in which to start.

Organizing To Do The Job

Perhaps the single, most important decision that was made by the college administration was that the job could not be done with the existing data processing staff. A new staffing proposal for a four member team (later increased to five) was approved and subsequently a reorganization took place that established a separate, long range, project oriented department whose responsibilities lay only in the future. It is amazing what can be done when last night's system failure is not your responsibility to fix. (Late in 1973 a Personnel/Payroll/Budget project was started with another new five member team.)

Committee Organization

The student registration project is directed by a five member committee including the registrars from the three campuses plus two individuals from data processing. Since the registrars are the only campus representatives on the committee, they are responsible for defining the system requirements with respect to their individual needs and for keeping their campus people informed about the progress of the projects and the changes that will occur because of it.

This committee has a very horizontal organization in that its members are nearly on the same administrative levels. Its personalities allow it to function more as a working group that meets to solve the on going problems than as a formal approving authority. The members are those who must define and live with the system and cooperation is enhanced because of these factors.

Contrasting this, the Personal/Payroll/Budget committee is very large (13 voting members) whose members consist of various levels of administrators. Those line managers who must live with the system are members but, because of the type of project, others were added to try to provide input to the system design from many levels.

Each project is functioning well under the guidance of its committee organization. The point here, then, is to choose that which fits the project rather than assuming that one method is better.

Summary of the Registration System

The new registration system design included these five major sections:

1. Admissions
2. Maintenance (Updating) of Student Personal Data
3. College Catalog and Master Class Schedule (creation and maintenance)
4. Drop/Add and Late Registration
5. Student Scheduling by Course and Major Time Block

The first four of these are all on-line functions while the fifth is, of course, batch.

The Basic System Design

Since the late sixties it seemed that Miami-Dade needed a system (from a computing and file organization point of view) that would allow the registration system to operate without regard to semesters or other artificial constraints. Admissions should be able to add a student to the system, the registrars should be able to enroll the student in classes as needed and the instructor should be able to post a completed grade to each student's record when the course requirements have been satisfied.

Thus emerged the concept of a central "pot" of data --- a data base if you will --- that would bubble and churn as information was added, changed and removed by those users who constantly interacted with the system. Since the "pot" was open ended, it would always reflect the latest status of all data and it could hold data for any given number of semesters either in the past or in the future. It could be asked periodically if any grades had been posted and then retire these courses to the transcript system while producing grade reports. This was the basic design of the system and many of these ideas can be done using the file structures that were chosen to support the system.

Organizing the Data Base

At the time that the registration system was being designed, there were several factors that precluded our choosing to select any vendor's "data base manager". These reasons included time to study the available systems, the hardware capacity to run them, the interface problems between systems and languages, the training problem and the money to acquire the package. In addition was the gut feeling that the choice of a data base manager required careful matching of a series of trade-offs whose parameters were not well defined. There did seem to be some viable choices that were open that existed using present day data access methods. This led to the choice of using several variable length indexed sequential files (ISAM) with one or more arrays to define the needed data elements. This structure is supported by COBOL, MARK IV and our teleprocessing control program.

The files that support the registration system are:

1. Student Master - contains basic biographical and statistical data along with a pointer to each active Student Course record.
2. Student Course - contains information that is term dependent and includes course requests, course assignments and other data.
3. College Catalog - contains one record for each course on each campus and is used to control the consistency of course offerings. (i.e., course title, semester credits)
4. Course Schedule - contains college catalog as the base segment with individual scheduling information added (days, times, rooms).
5. Section Pointer - used as a cross reference file to provide a linkage between the college catalog whose key is course abbreviation and course number and the course schedule whose key is sequence number (a unique 8 digit number).

File Integrity

Since the system design called for updating the only copy of the data base directly and this was our first effort of this kind, we took special care to ensure that we protected the integrity of the data base even at the expense of some other on-line considerations. Three special files were created (1) to indicate the file name, file status and record key, (2) to log a copy of each new or changed record in the data base and (3) to provide a quick, simple method for recreating a sequential output file.

The main problems with ISAM in the on-line environment occur during an add-new operation or during a record update that increases the length of an existing record. Both of these processes cause records within the data base to be moved within the prime data area or to the cylinder or even to the independent overflow area. If an event prevents the total completion of the movement of the data and the updating of the indexes, the file may appear to be intact and readable, but one or more records may have been lost.

Determining if a potential ISAM problem exists in our environment is a two step process. First, a record which is external to the file is written just before the update to indicate file name, record key and the exact function being initiated. Second, after the operation has been completed, the record is rewritten to indicate a successful operation. This record, the Update Log Record (ULR), may be interrogated at any time to display the current status of ISAM file changes. Since our system is essentially a single-thread program, only one ISAM file can be affected at any one time.

The ULR record also serves to maintain record counts for each ISAM file. Each time a transaction adds or deletes a record, the counts in the ULR are adjusted appropriately. The ISAM delete option is not used and record deletes are handled by the program that does the file reorganization.

Transaction Log Record

The second special file that was created is called the Transaction Log Record (TLR). This file contains a copy of each ISAM record before and after any changes have been made. In the case of an add-new to the file, it is, of course, only a copy of the added record.

This file serves two purposes: (1) as a method for detailing any data element changes to the file and (2) as a means of rebuilding the file in the event of an unrecoverable system failure. Since the date, time and file name is present in all of the records on the TLR, it is a minor programming problem to read the last file back up and do the appropriate merging with replacement to reconstruct the file.

Sequential Output Files

The Transaction Log Record is only one of several sequential output files that are created as the result of processing a transaction on-line. Typically, a file of this type is processed by a batch program after the terminal system is finished for the day. There are two problems in an on-line environment with using this type of file, one simple, one complex. If the file is defined to the system as being single buffered and unblocked, no data will be lost as a result of a system crash because each new record will be transferred to a direct access device as soon as it is written. This method is, of course, our shop standard.

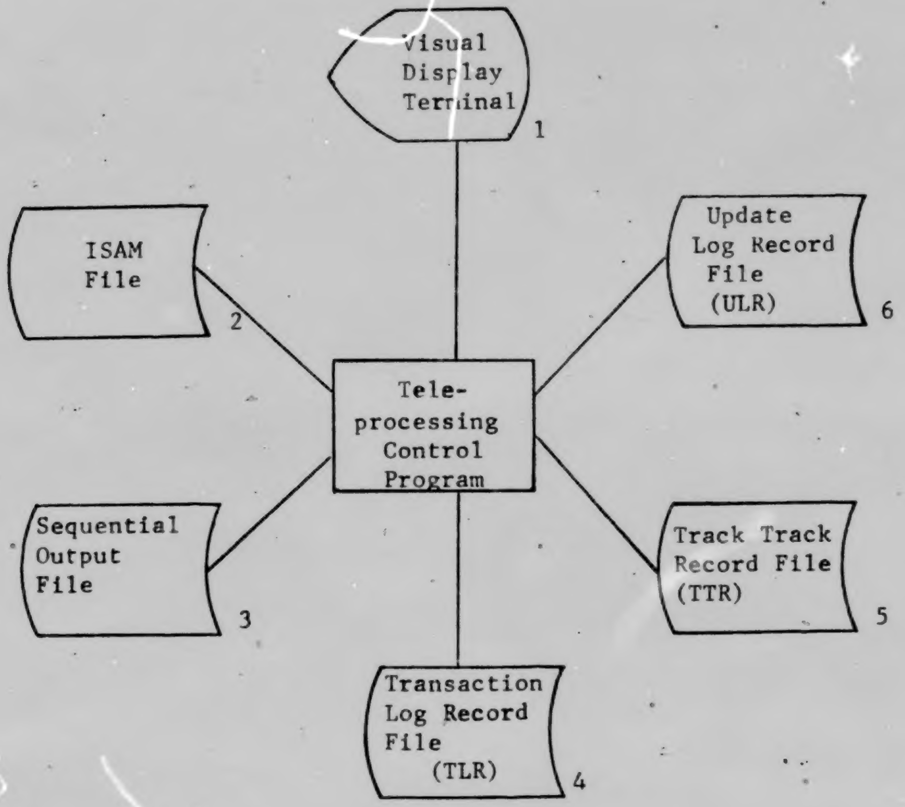
The complex problem occurs when, during a system failure, the system cannot close its files. In this case, no pointer information is available anywhere within the system to indicate where the next sequential record is to be written after the system has been restarted. A fast, simple restart procedure has been developed to control the problem by making use of the Note-Point feature of the Operating System. This feature returns to the programmer the relative disk track and record address of each record after it has been written on the disk. By keeping this address in a separate location (the TTR file) for the last two records written, it is a relatively easy programming task to initialize the system to point to the proper disk location upon restarting the system.

Track Track Record File (TTR)

The TTR file consists of an individual record for each sequential output file needed by the system. The TTR is updated after each time that a record is written to the sequential output file, thus making the TTR always current. To reopen the file properly, the TTR record is used to directly read the last record written; then, the TTR of the next to last record is given to the system and the last record is rewritten over itself. This works because the system updates the TTR before the next record is written. Consequently, the file can be opened and properly positioned for the next write by executing only a relatively few instructions.

Teleprocessing System File Overview

The following diagram illustrates the six components of our on-line system and the various files used.


Visual Display Terminal
1
ISAM File
2
Update Log Record File (ULR)
6
Tele-processing Control Program
Sequential Output File
3
Track Track Record File (TTR)
5
Transaction Log Record File (TLR)
4

1. Represents any key driven terminal
2. Represents any ISAM file with update and add-new
3. Represents a sequential output file created as the result of a user transaction
4. Represents a special sequential output file used to log transaction record for file rebuilding and change reports
5. Contains one record for each sequential output file to store the address of the last two records that were written
6. Contains one record to identify the file and record key of any ISAM file being updated

These files relate to each other according to the following chart.

	ISAM	SEQUENTIAL OUTPUT	TLR	TTR	ULR
ISAM			X		X
SEQUENTIAL OUTPUT			X	X	
TLR	X	(X)			
TTR		X			
ULR	X				

( ) Optional

As the chart indicates, ISAM (add new and update) files must use the TLR and ULR files while sequential output files may use the TLR but must use the TTR file.

File Backup and File Reorganization

The end of day processing is divided into two basic functions: transferring those "spin off" files from the teleprocessing system that feed the batch systems and initiating the periodic file backup up and file reorganization procedures that are necessary for security and throughput performance.

Two separate systems are now used to satisfy these needs, but both are variations of the same techniques.

The first system was merely to unload all of the ISAM files onto sequential tape files as a series of "generation data sets". The files are copied one after another onto the same reel of tape in order to make the operation as quick and easy as possible for operations. After all files are copied, the tape is rewound and the files are copied back on to the direct access files in order to remove records marked for deletion, to make sure all data is contained within the prime data area and to balance and update the record counts in the ULR file.

As time went on, the files spilled over onto the second tape volume causing some problems in terms of multi-volume handling by the operating system especially concerning the cataloging of data sets and the ability to keep track of the correct tape volume for the second reel of tape. Although the problems here were somewhat baffling, an adequate method for taking care of them was found.

When this procedure was first executed and the files were relatively small, the complete job could be run in 20 minutes or so, but as the Student Master file passed the 100,000 mark, the time approached 1 hour. Consequently we implemented a copy system to allow three disk generations which would cut out one half of the processing cycle. Under this method the time is reduced back to the 20 minute range and the system automatically changes the catalog entries to point to the correct volume. We plan on using this system except for Saturday night when we will copy back to tape.

Throughput Considerations

Response time is an important item in the design of any on-line system. And, while we would like to keep it in the 2-6 second range, there appears to be no real way to estimate what that time will be except in some very broad terms. Only after the programs are running and the processing starts will the true time be known. I think, though, that far too much time is spent worrying how to make the transaction coding and the teleprocessing manager as efficient as possible where the savings can be calculated in microseconds whereas much can be done by improving the hardware or the operating environment.

1. Increasing terminal transmission speed to 7200/9600 BAUD is relatively inexpensive.
2. Moving the polling operation from the central processing unit to a control unit can make a "remote" system appear to be a 'local' demand system.

3. Finding a method to leave some extra space on each track will significantly improve the speed that ISAM uses to process an increase in the record length or an add to the file.
4. Paying close attention to the locations of the data sets in relationship to themselves, separate disk spindles and separate channels will help minimize disk head movement during processing
5. Designing two configurations for the teleprocessing system, one for normal days and one for extra heavy days (making most commonly used functions core resident).
6. Creating a pointer table for the teleprocessing system that will let a core resident index point directly to the disk address of each function will cut out a directory seek.

Most of these methods are self-evident. It is very difficult to design and pay for a system that will handle all of the activity easily on the heaviest days. For instance, one of our campuses has historically completed the greatest part of their registration in three days just before the start of the semester. I don't believe that this system could support 4000-5000 per day even if the college could afford to pay for all of the needed terminals. The only thing that can be done is to spread the load over a longer period which in fact is advantageous to all. The basic system design supports around the year registration and controlling the volume means that the terminal operators can handle most of the work on a regular basis thus increasing machine utilization and improving operator effeciency. It is, however, a significant change to the operating mode of the campus.

Batch to On-line Conversion Procedures

The procedures used to transfer from a batch mode to on-line processing are not significantly different from any other conversion. In our case it was necessary to create a completely new file structure and organization. A file conversion program was necessary to transfer the available data from the old system into a format suitable for the new system and since we could not affort to replace the old system overnight, we chose to write a bridge program to move the data from the new file into the format of the old. This process means running duo systems but also means that we could continue to produce all existing reports without a massive reprogramming effort.

Perhaps the most significant difference is that there is less room for error in the programming. In a batch system, all input data is generally readily available for rerunning in the original sequence in one place. The worst that can happen is to restore a copy of the file before processing, correct and rerun the program. Since the on-line system is both random and remote by nature, the problems of correcting mistakes can be cumbersome if not impossible. However, the ability to see exactly what the data is now, typically makes error detection and correction easier.

On-line Testing and Maintenance

The file structure, the operating procedures of the computer center and the capabilities of the teleprocessing control program all interact to form the on-line testing and maintenance environment. Basic planning in these areas will pay good dividends as the project comes alive.

Test files can be made separate from the real files and can be defined to the teleprocessing systems so any experimentation on-line cannot damage the real data. Generally, there are two disadvantages to this approach; (1) there is extra overhead in opening these extra files and (2) two sets of teleprocessing programs must be maintained in order to read and write the correct files. Additionally, keeping track of the correct version is a problem in itself. Sometimes test data sets are smaller than the real ones thus reducing the possibility of finding exceptional conditions.

A method for testing that is especially useful for testing on-line is placing the test data in the same file as the regular one but changing the key slightly. This is very easy in our course file that is keyed by a composite of year/term and sequence number. An existing term of data can be copied changing the term to one far in the past. This also means that the programmer may use the on-line system to easily modify the test data in order to validate new programs.

There is significant change in the closed shop testing procedures for batch and on-line testing. Unless the capability for complete on-line testing and maintenance exists, some compromise must be worked out in order to give the teleprocessing programmer access to the terminal hardware and notification to the programmer when the batch update run has been completed. Presently, we are giving our TP programmers limited access to the computer room for testing purposes. This is not too hard to control since only 1 or 2 are actively debugging at any given time.

It is especially important that some method be worked out to notify the programmer when the update is complete so that a test shot does not require two separate efforts to complete.

The selection of the teleprocessing control program, especially its testing and debugging facilities, is a very important consideration. Ours offers two distinct features in both on-line testing and in the use of its card simulator.

Since our teleprocessing control program has its own native language (an RPG, specifications type, if you will) it is almost impossible to bring the system down because of bad coding. Secondly, it takes only a very short time (5 to 10 seconds of clock time will handle most) to compile and catalog a module for testing. As soon as the module is placed in the system library, it is ready for testing. The interpretive system is such that the transaction will continue to execute until completion or until a serious error forces its termination.

The card simulator provides a complete trace of the program's execution instruction by instruction so its execution logic may be verified by the programmer. Any time an instruction cannot be executed properly, a code is displayed to describe the problem. The ability to snap dump input/output areas, user work areas and certain other control tables provides debugging flexibility. A complete dump of the region is available after all processing is ended.

All of these capabilities make the teleprocessing system easy to use and stable - two important factors in our environment.

Impacts of Project Design and Implementation

Changing from batch to on-line provides an opportunity to review both the organizational structure as well as the job responsibilities of those offices involved. This review process in combination with the new computing capability should cause some of the following things to happen:

1. Job functions will become better defined both as to what it is and who should do it.
2. Responsibility for data capture, entry and maintenance will be specified element by element.
3. Functions may move from one department to another
4. The system becomes supportive of a multi-department operation each having access to the data it needs. Separate files disappear - the data base emerges.
5. User becomes responsible for his data

A Registrar Reflects on the Change to On-line

Recently, I visited one of our registrars to get his comments on what the change to on-line meant to his operation. I inquired as to both the positive and negative aspects of the transition.

His first reply was that he felt that is was difficult to identify any significant problems other than the obvious one of system down time. In an on-line environment, the problems of the computer center become the problems of the user - they are readily and immediately apparent to all. This is somewhat of a change from a batch orientation since many of the shorter computer breakdowns are never seen by the user.

Summarizing the other negative aspects, he cited:

1. The expense of the hardware and programming.
2. Too stringent of control in which office can update which elements. Rather than send the student to the appropriate office, the change is not made.
3. A reluctance to remove data from the on-line file because of the loss of the update capability.
4. The computer response time tends not to handle peak loads.

5. The reluctant loss of source documents.
6. The possibility of having to re-enter yesterday's work if both the system fails and the file rebuilding system fails.
7. After the initial success, everyone wants on-line.
8. The registrar's office was reorganized as a result of going on-line.

On the other hand, he was easily able to give the following positive aspects.

1. Eliminates source documents.
2. Eliminates transcription and retranscription errors
3. Eliminates errors that only the student (since he is often standing by the terminal) can correct.
4. Reduces errors because only individuals who are totally familiar with the data handle the input
5. Edits all data at the source
6. Eliminates all the "middle men" in the batch update cycle.
7. The registrar is responsible for his data
8. Retraining was relatively easy.

Variable Length Records and the COBOL Compiler

Version IV of the IBM OS ANS COBOL compiler supports only variable length, sequential files; consequently, the first thing our systems programmer had to do was to write ISAM support in assembly language so that it could be called from a COBOL program. This was not an exceedingly complex problem and, except for the normal debugging process, went rather smoothly.

The main problem shows itself when variable length records, whose record description contains the OCCURS DEPENDING ON clause, are moved from one area to another. The compiler mistakenly assumes that the record size should be calculated using those fields that are the object of the DEPENDING ON clause in the receiving area rather than in the sending area. Our original solution to this problem was to move those control fields from the sending area to the receiving area as a series of individual moves after which the whole record was moved. While this method worked, a better one was needed.

In order to avoid using name qualifiers it is advantageous to have two record descriptions to define the same record with only a slight, standard change. Our standard evolved as follows.

```
01   SCR-I-STUDENT-NUMBER
      where SCR is a 3 character abbreviation for the record
            (Student Course Record)
             I is "I" for input, "O" for output
      STUDENT-NUMBER is a descriptive name for the item
```

By defining the COBOL record properly, variable length records can be moved by moving the entire base of the record with the first move and then moving all of the variable part with a second move. The COBOL record description entry to do this looks like this.

```
 01   SCR-I-STUDENT-COURSE-REC-VNI.
      05    SCR-I-BASE.
┐             10  SCR-I-BASE-ROW-AREA.
┐                 15  SCR-I-OS-CONTROL-BYTES.
┐                     20  SCR-I-OS-RECORD-LENGTH    PIC S9(4)   COMP SYNC.
┐                     20  SCR-I-OS-RESERVED         PIC S9(4)   COMP SYNC.
              10  SCR-I-BASE-RECORD-AREA.
                  15  SCR-I-OS-DELETE-BYTE          PIC X.
                                  .
                                  .

                      remainder of base area
                                  .
                                  .
                                  .
                  15  SCR-I-ARRAY-CONTROL-CODNTERS.
                      20  SCR-I-REQUEST-COUNT       PIC S9(4)   COMP SYNC.
                      20  SCR-I-COURSE-COUNT        PIC S9(4)   COMP SYNC.
                      20  SCR-I-OVERLAPPING-CRSE-COUNT
                                                    PIC S9(4)   COMP SYNC.
      05    SCR-I-ARRAY-AREA.
              10  SCR-I-REQUEST-ARRAY.
                  15  SCR-I-STUDENT-REQUESTS OCCURS 0 TO 25 TIMES
                                DEPENDING ON SCR-I-REQUEST-COUNT
                                OF SCR-I-STUDENT-COURSE-REC VNI.
                      20  SCR-I-REQUEST-INDEX-LOCATION
                                                    PIC S99     COMP-3.
                                  .
                                  .
                                  .
                          remainder of request array
                                  .
                                  .
                                  .
              10  SCR-I-COURSE-ARRAY.
                  15  SCR-I-STUDENT-COURSE OCCURS 0 TO 40 TIMES
                                DEPENDING ON SCR-I-COURSE-COUNT OF
                                SCR-I-STUDENT-COURSE-REC-VNI.
                      20  SCR-I-COURSE-ID.
                          25  SCR-I-COURSE-ABB      PIC XXX.
                                  .
                                  .
                                  .
```

```
                              remainder of course array
                                        .
                                        .
                                        .
           10  SCR-I-OVERLAP-COURSE-ARRAY.
               15  SCR-I-TIME-EXCLUSIONS OCCURS 0 TO 8 TIMES
                             DEPENDING ON
                             SCR-I-OVERLAPPING-CRSE-COUNT OF
                             SCR-I-STUDENT-COURSE-REC-VNI.
                   20  SCR-I-CE-DAYS           PIC X(7).
                                        .
                                        .
                                        .
                              remainder of overlap course array
                                        .
                                        .
```

To move the record from an input area to an output area requires these statements.

```
     MOVE SCR-I-BASE TO SCR-O-BASE.
     MOVE SCR-I-ARRAY-AREA  TO  SCR-O-ARRAY-AREA.
```

The base record is divided into two 10 level entries in order to facilitate the movement of data between files supported by COBOL (whose length does not include the record description word) and that which our systems programmer wrote to handle variable length ISAM. Note that the four cards with the "not sign" in card column 7 are changed into blanks or asterisks depending upon which file support is needed.

These COBOL record description construction standards along with naming convention standards should provide an excellent base for permitting file and program cross referencing at the data element level. A machine supported, automatic system of this type would make possible the dream of maintaining documentation to answer the question "What programs and reports will be affected by changing this data element?"

Looking Back

Standing near the end of the project and looking back, there are many small things that I would have done differently - fortunately no large ones. Some of these changes are:

1. A formalized data dictionary should have been completed.
2. COBOL record description standard should have been refined more.
3. Perhaps a little more data redundancy would have been permitted to cut down on input/output processing time.
4. The file recovery control system would not have been so exhaustive.

To summarize, the most important thing that can be said about a project of this size is to proceed slowly and let the system evolve. Large, on-line systems are so complex and interrelated that all parties concerned especially data processing and user must be convinced that they are alive and well and ready to go. This means that target dates, although meaningful and necessary for management planning, should not be met if there is any doubt concerning the ability of the system to perform. The penalty for failure is too great.

FUNDAMENTAL BENEFIT-COST CONSIDERATION FOR SYSTEM DESIGN: THE MIS INFORMATION/ACTIVITY MATRIX

Robert Harris
Assistant Director
Division of Management Information
and Telecommunications Systems
U. S. Atomic Energy Commission

MIS INFORMATION/ACTIVITY MATRIX

Introduction

There is a need to know, in approaching an MIS development proposal, what is missing or unsatisfactory in the current state of information systems affairs, and how a proposed MIS development would solve or reduce the problems, or achieve an opportunity. The extent to which a proposal responds to the problems, and the significance and number of problems solved, or contributions of opportunities, should bear heavily on the GO/NO GO cost benefit based decision and on the establishment of priorities. At present we describe these matters in the Statement of User Requirements, but, inevitably, they become enmeshed in other matters which the document covers. The MIS Information/Activity Matrix brings clear focus to the value issues and the significance of the proposal.

Scope

This paper describes a technique for presenting a statement of how a user is inhibited, at present, in the performance of his managerial or administrative functions by information problems or inadequacies. It also shows that the same technique can be used to indicate how a proposed MIS would affect the user's information problems and if it would enhance, through improved information processes, his performance of management or administrative functions.

Discussion

A. General

In the AEC, one of the responsibilities, in connection with MIS, of Managers of Headquarters and Field Office programs and functions is the assurance that AEC planning and control activities are adequately supported by cost effective MIS. The Guidelines note that information contributes to programmatic or administrative effectiveness if it facilitates:

1. Recognition of existing, incipient or potential problems or opportunities
2. Analysis for problem solving or planning
3. Decision making
4. Control of the implementation of decisions and plans

5. Performance of operational, functional and logistical activities

6. Responding to external demands for information

- and the more an MIS development contributes to these functions the more is its value, and this is the best way, in relation to costs, of measuring whether it should be accomplished and if so its priority for development.

An MIS is valuable in the support of management functions to the extent that it produces supporting information at a lower cost than alternative means.

The information problems stipulated by the Guidelines, which inhibit the named management functions, and which an MIS development may correct or ameliorate, are:

1. Unavailability of information

2. Inadequacy of the quality of existing information

3. Inadequacy of the processing of existing information

4. Lack of timely accessibility of existing data

5. High cost of present information handling processes.

The attached sample MIS Information/Activity Matrix presents the proposition that there are known management functions which are supported by information, and that there are known information faults or inadequacies which act to inhibit the effective exercise of these management functions.

Comparison of a number of these Information/Activity Matrixes, written for different system proposals, will show judgementally, that some "do more" for users than other proposals, and therefore have more "value". The final judgement process must assess the relative importance of proposals to the mission of AEC, or to management. Those proposals that "do more" can be considered more desirable (i.e. having a higher payoff) and should receive higher priority consideration for implementation. Of course, likelihood of success, consequential effect, etc., are among other matters to be considered in reaching decisions on priorities; however, these matters are not within the scope of this paper, which is focused on the expected merits of proposals that are to be judged in relation to costs.

B. Specific

I. There may be a number of different occasions when iterative preparation of the Information/Activity Matrix would be useful:

1. As an initial statement of "what's wrong?" In no case should this be done without user participation, since his participation has two benefits:

    a. As an "icebreaker," and to be sure that he can make his problems known (and so that it can be established that problems really exist).

    b. As a medium for his expression of his desires for service. Peripherally, but importantly, some users, who have been kind enough to work with us in the earlier stages of our thinking, have commented that the thought provoking nature of this analysis has been helpful to them in understanding their problems. We have found that these users have tended to concentrate more on the substance than on the technology of the matter at hand.

2. At the completion of General Design, to be compared with the initial Matrix, to insure correspondence between user need and designer's response (or to establish the fact and specifics of non-conformance).

3. At the time of any major system change which, it is suspected, may result in change of system response to the user's original statement (and to express a change in user desires, if required), for comparison with earlier versions.

4. At the time of a system audit, as a comparison against earlier versions of the Matrix to determine whether the system, as developed and operating, is still responsive to user need, whether user need has evolved into a new pattern, etc., and thus to achieve "early warning" of improvement requirements.

II. In preparing the Matrix, it is necessary to recognize that its components are not separate, unique matters, but that they blend and interweave with each other. Nevertheless, some descriptive material will help to avoid confusion:

1. Unavailable Data. Data is unavailable if:

    a. It plainly does not exist or is not now being collected

b. It is available, but this fact is not known to the user

c. It is available, but classified; and the user is not appropriately cleared

d. It is available, but at a cost the user cannot afford.

2. Inadequate Quality Data. Data is of inadequate quality if:

   a. The SPEED of the regular processes of the system including production of routine outputs is unsuitable

   b. The ACCURACY of the data is unsuitable (i.e. data exists but is inaccurate, unedited, in some way suspect, or invalid)

   c. The FREQUENCY of the data is unsuitable (i.e. it does not cover appropriate time periods; cannot be had monthly, but only quarterly, for example, or the present system can only be run on some unrealistically rigid or constrained schedule).

   d. The SCOPE of the data is unsuitable (i.e. the data cover too narrow -- or too broad -- a range to allow precise "viewing" of a particular matter).

   e. The SELECTIVITY of the present system is unsuitable (i.e. the system does not permit retrieval of only what is desired, rather it provides a bulky output in which the user must locate the particulars he wants).

   f. The ANALYTIC ability of the present system is unsatisfactory (i.e. there is no, or unsuitable, analytic process to raise outputs above the "stream of facts" level).

   g. The SECURITY of the present system is unsatisfactory. At present, automation of MIS processes is not usually helpful, but we believe this will not always be the case. At least, the new proposal should be inspected for security problems; at best, it may be possible to improve Security, particularly in large volume situations.

3. Inadequate Processing of data

This area is concerned with lack of processing in the sense of ordering, sorting, collating and tabulation, and with mode of presentation. The goal of MIS employment is useful information for a manager or managers; it must be remembered that management styles vary and that MIS must be responsive to such variation, in addition to having valid, reliable and stable internal processing.

4. Untimely Accessibility

   This refers to the response speed of the system to ad hoc inquiries rather than the routine processes mentioned in II. 2. a. above. If concurrent access to data from various sources is required and one is untimely, the process is untimely from the manager's viewpoint.

5. High Cost

   This means "high" in terms of what the user can continue to afford (dollars, machines, or people).

III. The notational method for filling out the Information/Activity Matrix may vary from one situation to another. There does not appear to be particular merit in the idea of uniformity, but rather the goal should be meaningful communication. The following should be considered:

1. Words such as "HIGH" or "LOW" or blank space to indicate the degree of intensity of user's opportunity or problem.

2. The word "NEGATIVE" to indicate that a proposed system responds less well than a present system in some particular. This will usually represent the result of a trade-off (e.g. timeliness vs. accuracy).

3. References to text in an attached document (usually the Statement of User Requirements).

Summary

An analytical technique is desirable to focus attention on the management payoffs of an MIS development. To the extent that management is inhibited by information problems in discharge of its functions, it is proper to consider MIS development as a solution. The degree to which MIS facilitates management functions should bear heavily on the priority assigned to particular projects. While it is understood that other elements of decision about priorities exist, it is the responsibility of systems staff to provide as much analytic input to the priority decision as possible.

One suitable way to provide analytic input is the Matrix described in this paper. This Matrix can also be used at other times in the development cycle and during the life of an MIS.

Proposal Name____________________ **Date**____________________

MIS INFORMATION/ACTIVITY MATRIX

	PROBLEMS				
Management Activity	Unavailable Information	Inadequate Quality Information	Inadequately Processed Information	Untimely Information	High Cost Informatio
Recognition of Management Problems or Opportunities					
Analysis, Problem Solving, Planning					
Decision Making					
Control of Plan and Decision Implementation					
Performance of Activity (Operational Functional, Logistic)					
Response to External Demand					

Prepared By:

DESIGNING A BATCH SYSTEM FOR TRANSFORMATION TO A TELEPROCESSING SYSTEM

Judith Campbell
Associate Director
Management Systems & Analysis
Cornell University

J. Paul Wetterau
Senior Systems Analyst
Management Systems & Analysis
Cornell University

Designing a Batch System for Transformation to a Teleprocessing System

J. A. Campbell

J. P. Wetterau

Cornell University, Ithaca, New York 14853
December 2, 1974

I. Introduction

The purpose of this presentation is to detail some of the design considerations which should be taken into account when planning a batch system which will be converted to an on-line system, or at least to allow on-line inquiry and update.

At Cornell there has been interest in the development of administrative teleprocessing capabilities for at least five years. Some background information on how we arrived at our current level of knowledge and technical capability provides an interesting and important prologue to the design and development of the Motor Vehicle System. The material will therefore be presented in two distinct parts, this historical background of administrative teleprocessing at Cornell and the reality of the decisions which were made and why they were made in the design of the Motor Vehicle System.

Clearly, the first problem to be addressed in designing an interactive system is data. How can it be organized for easy retrieval of needed items? How can the cost of storing it on-line be justified? What new security considerations come into play? At Cornell the decision which was made, in 1970, was that rather than provide a few small data bases and inquiry modules for selected users, the university would develop an overall management information system and develop on-line capabilities as a logical outgrowth of the availability of data. Over the next four years the reality didn't quite develop in accordance with the original plan.

The first step taken by the University was to evaluate several available data base management systems. After a serious study of systems available at that time the decision was made in March 1971 to install IBM's IMS, in large part because it appeared more readily adaptable than its competitors to TP systems. The decision made at that time was that due to the level of expenditure involved IMS Batch would be installed first, to be followed within six months by IMS/TP capabilities using IBM 3270 CRT's.

The first system put up under IMS was a consolidated Payroll/Personnel System which had not originally been designed for IMS implementation. This system which was installed in September 1971 without any substantial or organized prior testing did major damage to CU's timetable for the implementation of IMS/TP as the costs and run times of the system far exceeded what had been expected.

Simultaneously with the design and programming of the Payroll/Personnel system another IMS system was designed and installed for the Office of Academic Funding. The system was installed in August 1971 with little complication and the first trade-off of the MIS concept was realized when data on faculty was able to be extracted from the Payroll/Personnel system rather than GCA maintaining this data and attempting to keep it updated. Everything was, however, still strictly batch.

During this entire time span the University Registrar and members of various college offices had been hearing about the on-line systems which were being installed elsewhere. Since Cornell's student system dated back to the 1401, requirements for a new Student Registration and Scheduling System including the use of data base concepts and on-line inquiry were developed. At this time Cornell studied the totally interactive system being developed at VPI and talked to several outside vendors of registration and scheduling software, as well as looking at systems being used by other universities. A decision was finally made that CU was not ready for a totally TP system including on-line update and a contract was signed with an outside vendor to provide an IMS based system with limited on-line inquiry capabilities. Based on the time frame agreed upon the University placed an order with IBM in November 1972 for the delivery of 19 3270 CRT's to be delivered over a period of 18 months for use in the colleges offices.

As a first step in preparing for this on-line capability the IMS/TP software and one 3270 were ordered by CU in January 1973 for March installation. At that time the University was processing on a heavily loaded IBM 360/65 with OS/MVT/HASP which had been "tuned" locally. As a result IMS/TP was installed only in a test system to operate in a stand alone mode until after the inquiry software was tested.

During March and April of 1973 I was asked to prepare a proposal on the outlook for administrative teleprocessing at Cornell. After evaluating the level of user commitment and preparedness, the University's financial position and the overall cost of large scale installation of TP I found myself very reluctantly recognizing the realities of the situation and therefore recommending that the time for TP at Cornell was not now. In October of 1973 the 3270 and the IMS/TP software were returned to IBM and the other terminals were cancelled. Unfortuntely the software vendor's inquiry module originally due had not yet been delivered but CU had successfully developed and tested an inquiry module under IMS/TP.

In addition to the cost factors involved in teleprocessing, our selling job was further complicated at this point by the increasing concerns about the security of data in general, and on-line data in particular. The general level of concern on data security has risen significantly in the last few years and at Cornell we had our own incident to fan the flames. Student operators working for the Office of Computer Services acquired data from the OCS accounting system, and then publicized their accomplishments. Their acquisitions included all the authorized access codes including those for Payroll, Student Records, etc. Needless to state our qualified technical assurances were not very reassuring.

Up to this point this sounds like the saga of why Cornell doesn't have TP yet. In September 1973 a problem developed which turned out to be a significant advantage for advocates of on-line systems. Cornell's administrative computing support group, the Division of Management Systems and Analysis (MSA), moved into "new" quarters in Rand Hall. This allowed consolidation of the entire staff which had previously been located at two sites separate by five miles, but there was a catch. The programmers had previously been located at the computer center, now the budget would not allow for an RJE to be installed in our new location.

Just about this time the Office of Computer Services (OCS) installed TSO. After some discussions it was decided to order two Hazeltine 2000 CRT's and experiment with having the programming staff enter jobs, update code, etc. in what was really, at that time, a sort of on-line batch. Despite the serious limitations of TSO, as installed at Cornell, and some interesting interface problems with the Hazeltines this turned out to be a good move. We now had some CRT's, and some people trained in their use. We also had a Datel terminal for APL. At this point we began to explore the potential of doing on-line systems either in APL or in PL/1. Since Cornell does not have APL-SV or APL Plus the inability to access data bases made the potential use of APL fade rather rapidly. On a visit to SUNY Binghamton which was writing administrative modules using APL-SV we also had reaffirmation of our concerns about the problems in involved documenting and maintaining APL code. We did attempt to develop a very small system in APL for MSA billing to users but eventually aborted the effort.

At about the same time we began to experiment with PL/1 under TSO as an interactive vehicle. With the extensive capabilities of the PL/1 optimizing compiler, and the work of very imaginative staff our early efforts were encouraging. Now all we needed was a willing guinea pig.

The Cornell Office of Transportation services is a related activity of the university. Their existing "data processing" consisted of tub files which were clerically interogated to provide information regarding motor vehicles registered by students and employees. Earlier discussions with this office regarding replacement of their tub files by a batch system had met with no enthusiasm, they felt that the turnaround times involved would provide no improvement over their old "system".

The decision was made in early 1974 to approach OTS and explore their interest in an on-line system, with batch processing as a first step. Several things made them an ideal potential customer - their system was relatively small, they were a "hard dollar" user rather than being budgeted from allocated funds, and they were co-located with MSA in Rand Hall so it would be possible to easily assist in the transition to what we hoped would be the University's first production TP system. The Director of OTS was very enthusiastic and requested that we develop a proposal for such a system.

II. Design Considerations Involved in the Executive Staff Presentation

Once tentative approval had been obtained from the Office of Transportation, the MSA Development Group then worked on preparing a presentation for the Cornell University Executive Staff describing the current problems faced by Traffic Bureau personnel, and the recommended system design considerations to meet these needs. These design features included the security precautions designed into the systems into order to allay some of the security concerns of the executive staff. These had been caused by some of the events which have been described earlier.

Many factors were considered in the over all design of the system, (as shown in Figure 1), labor savings, more up to date information, quicker retreival through on-line access, and data security. It was felt that the Executive Staff presentation should emphasize the use of Management Information System (MIS) concepts to reduce clerical effort, data redundancy and duplicate data collection and maintenance, as well as the security aspects of the system. We also planned to discuss the use of new data input techniques (e.g., optical scanning to cut costs and shorten the data prep time).

A. Use of MIS Concepts

We used MIS concepts through the linking of the Motor Vehicle System with the Payroll/Personnel and Student Information Systems, the official university employee and student records. Extract programs were used in order to obtain address, home department or College, and special parking permit eligibility information for the Motor Vehicle System. This design feature thus eliminates the need for the Traffic Bureau personnel to also collect and maintain this information, and also helped insure data uniformity and accuracy. The extract programs were to be run on the same time schedule as the updates to the two feeder systems to insure that the data on the traffic file was timely.

B. Security Concerns

Since security would be foremost in the minds of our audience, with this system being proposed as the University's first on-line administrative system, and also because we were, at least in the minds of the uninitiated tieing together administrative systems in order to eliminate double collection of data we knew that the main thrust of our presentation had to be to allay these fears and misunderstandings.

1. Security of Outside Feeder System Data

Since we were using both the Student Data Base and the Payroll/Personnel (P/P) Data Base both of which carried sensitive data the fear existed that some one who had authorized access to the Motor Vehicle System via a CRT terminal could go backward through our linkages and therefore obtain access to the sensitive Payroll data (e.g., employee salaries) or sensitive student information (e.g., student grades) and retrieve or alter them. We therefore designed our interface (See Figure 2) so that a Social Security Number/License Number Cross Reference file which was produced and maintained by the Motor Vehicle System File Maintenance program would be used to access the Payroll/Personnel and Student Information System Data Bases.

A missing records file was also created by the maintenance program in Social Security Number sequence to identify those records on the P/P and Student Data Bases of individuals who had just registered their vehicles and were therefore missing biographic information. Change dates on both feeder data bases were also interrogated to determine updated information to be added to the Motor Vehicle Data Base. Since the Cross Reference files were sequenced by Social Security Number which isn't carried on the Motor Vehicle Data Base, there was no direct way to use the Cross Reference files to find out information about someone on the Motor Vehicle Data Base which is kept in License Number sequence and then use this information to enter the Payroll/Personnel or Student Data Bases which are ordered by Social Security Number.

As a further precautionary security measure it was also decided that the maintenance programmers who had the authority to access the Payroll/Personnel and Student Data Bases respectively would be the ones writing the extract programs rather than members of the development team. The production controllers having cognizance over those two areas would also be the ones assigned to running the extract programs.

Once the method of obtaining data in the most secure fashion was designed and additional security procedures established as previously described it was then necessary to negotiate between the Traffic Bureau (the System User) and the custodians of Payroll/Personnel and Student Data Bases to determine what data elements from those Data Bases were needed directly by the Traffic Bureau in order to properly run their operation. An example of the compromises our office worked out in order to protect data on the two feeder data bases to the greatest extent possible was the case of 'U' permit eligibility.

The 'U' parking permit is the only permit that enables faculty and staff to drive onto the center of campus and park in authorized parking lots on center campus. All other parking permits are valid in peripheral lots and are denied access to the center of campus.

The privilege of obtaining a 'U' permit is extended only to "exempt" staff and faculty employees earning over a specified salary and also non-managerial employees who have 25 years of service with the University. Incidently this privilege costs the "lucky" employee that qualified $108 per year if he opts for a 'U' sticker. The Traffic Bureau asked that Salary, Fair Labor Employee Status, and Hire Date information be supplied by the custodian of the Payroll/Personnel Data Base, who balked because of the sensitive nature of some of this data. MSA then had to work out a compromise to the benefit of both sides. The solution was for the extract program which accessed the Payroll/Personnel Data Base to look at the three fields involved and create a "yes" code in a flag field to be passed to the Motor Vehicle System if the employee was eligible for the 'U' permit thus satisfying the needs of both the user and the data base custodian.

2. Data Secutity within the Motor Vehicle System

In addition to the security precautions mentioned which are intended to protect data carried in outside files, one important security precaution was designed into the prospective on-line Motor Vehicle Traffic Data Base. It was decided that since this data base would be on-line and a vehicle registrant's data could be viewed on a CRT in the Traffic Bureau Office, the registrant's social security number, due to its sensitive nature, would not be carried on the main Motor Vehicle Data Base. Of course it was necessary to carry the social security number in the system so that biographic data could be accessed from the other data bases previously described. We therefore adopted the off-line Cross Reference file concept again and designed a LIC NO/SSNO X-reference file which would carry the registrant's social security number only. This record would be created on the X-reference file when the Registrant filled out his initial application. As a further precaution the social security number field is not included on any screen formats, or any of the batch lists of vehicle registrants data which were to be used by the Traffic Bureau. In this manner we felt we could show the executive staff in a dramatic fashion how one of the most controversial and sensitive data elements (social security number) was being secured by the design of the system.

III. Results of the Executive Staff Presentation

When the broad systems proposal was presented to the executive staff two basic actions were taken, one of which was hoped for, and the other one expected although not hoped for. Due to the known opposition because of security concerns to on-line administrative systems our presentations outlined an option of first producing a batch system which would take advantage of the MIS concepts, data input techniques, and more timely data updating and then phase into the on-line file query and updating features desired by the Traffic Bureau.

The executive staff and the user felt that both the batch and on-line design considerations were sound and protected the university adequately from a security stand point. However, due to the political nature of on-line teleprocessing at Cornell the two decisions which were reached in this area were:

1. Approval was received to go ahead and develop a batch processing system which would produce weekly at least a listing in license number sequence of virtually all data carried in the system on a registered vehicle. This would replace the current tub file and would contain all biographic data on the driver and owner as well as parking permit related information, vehicle data (e.g., color, body type, etc.) and parking violations data. Two cross reference lists were also designed to facilitate retrieval.
2. It was also decided that the on-line query and updating module would be designed and programmed by MSA as a shelf item to be introduced when the batch system had been shaken down and the political climate was right.

IV. General Systems Design Considerations

Since the decision had been reached to proceed with the batch system with planned future evolution into a teleprocessing system the system's design had to include the "hooks" for efficient and secure on-line processing especially with regard to the files used and the design of these files, in particular the Motor Vehicle Data Base. Also additional teleprocessing security features were designed into the system in an attempt to help overcome all residual resistance, and the design of the system was also effected by outside factors. The overall design features which evolved are described below.

A. Batch File Maintenance Program Design

When we designed the batch file maintenance module we added the capability of also updating the on-line update data base. (See Figure 2) The update data base will carry all file update transactions that enter the system on-line during the previous processing period (8:00 A.M. to 5:00) and will be a mirror image of the on-line Motor Vehicle Data Base, except that it will only carry records of Motor Vehicle registrar's who have had activity during the last processing period. The Data Base itself will be updated in batch mode during the second or third shift. This delayed, batch method of updating the data base was chosen to protect the integrity of the Data Base against system crashes and thus the frequent need to restore the file.

As is shown in Figure 3 any on-line queries against the file will first be directed against the Motor Vehicle Data Base and if a flag is present indicating activity during this processing cycle the inquiry will be automatically switched to the update file. All on-line updates will refer to Master File and then be copied to the Update Data File in the updated condition.

A further need of the on-line system which has been designed into the batch system was the On-Line Name/License No Cross Reference File. This file will enable the Traffic Bureau personnel to inquire about an individual's motor vehicle by keying in his last name. The on-line processing module will then present a screen containing the full names and license numbers of all vehicles who's registrant's have that last name. Placing an (x) next to the individual's name whose record is desired will then cause the master file to be referenced. If the master file has been updated, then the Update Data File will be displayed. The format for the on-line display of the data base is shown in Figure 4. If a registrant has more than 5 violations (tickets) on a particular vehicle an additional screen will be generated upon request.

B. Data Base Design Considerations

The Motor Vehicle Data Base was designed as a hierarchical file particularly due to the need to secure certain types of data carried on the file from unlimited access and updating by Traffic Bureau personnel with a need to access less sensitive data. Therefore, the data base was structured as shown in Figure 5. Thru use of access codes a clerk can be denied the ability to update the parking violations segment on-line. The updating of this segment can be restricted to batch input only, or on-line access can be given to supervisor who is the only one authorized to use a certain terminal.

Since financial considerations precluded the use of IMS-TP as mentioned earlier we utilized the capabilities of the PL/1 Optimizing compiler linked up with TSO to provide on-line access to the hierarchical Motor Vehicle Data Base. The use of the "refer" option available only under the PL/1 Optimizing Compiler enabled us to create a multiple number of variably occurring segments thus simulating an IMS hierarchical data base.

The use of a hierarchical structure will also ease later conversion to an IMS data base should the leasing of IMS-TP be approved by the University at a later date as an intergal part of the Cornell Management Information System Concept. This would then cut down on the necessity to write future conversion programs.

C. Design Considerations of Supporting Files

As we mentioned earlier in discussing the security precautions designed into the system when it was presented to the executive staff, the use of an off line License No/Social Security Number Cross Reference File has been provided for in the batch system. This file will contain the vehicle registrant's social security number so that it does not have to be carried in the on-line Motor Vehicle Data Base. This data base will be up throughout the entire first shift when the teleprocessing module is installed in production mode. By using the License No/SSNO X-Ref. File in the batch system no file design changes are needed to protect the accessing and identification of student and staff social security numbers when operations switch to an on-line environment.

D. Additional Design Features of the On-Line Module

In addition to the features included in the batch system to support to conversion to teleprocessing, the real time file inquiry and updating module has been designed to contain an additional level of security for the system.

A CRT operator in the Traffic Bureau will have the ability to access the Motor Vehicle Query and Update Module through their terminal using the access codes provided. Once the operator has logged onto this module he will only be able to access the Motor Vehicle Data Base. If he or she would attempt to "crash" the Motor Vehicle maintenance program in order to obtain access to another data base which was on-line at the same time (e.g., Payroll/Personnel Data Base) an initial warning will be given over the screen telling the operator to cease and desist. A repeated attempt will then result in the on-line module software generating a "log off" command thus taking the malevolent operator off the system.

The on-line module will also contain a feature which will access the operating system clock, and can thus be programmed so that the on-line module can only be activated during the first shift (8:00 A.M. to 5:00 P.M.). This will prevent Traffic Bureau personnel from coming in during evening hours in order to "play" with the system in an attempt to gain illegal access to other data.

E. "Real World" Problems Effecting the System Design and Development

Although the system's overall design stayed intact throughout the development phase of the system, several changes were forced in the method of data input due to external forces.

In addition it was discovered during the detail design of the module interfacing with the "old" Student Information System that certain address data on the student data file was incomplete and therefore, could not be used in the Motor Vehicle System. This discovery necessitated a modification to the Vehicle Registration form to enable the students to supply local address information. This problem will be corrected when the new Student Records System is installed, and this duplicate collection of data can be eliminated in the future.

This covers the major design considerations involved in this system. Many of these features should enable this system to process efficiently and with little conversion effort if and when the decision is made by Cornell's top management to evolve into the administrative teleprocessing environment.

FIGURE 1

SYSTEM DESIGN CONSIDERATIONS

SECURITY

LABOR SAVING TECHNIQUES

- OPTICAL SCANNING
- TURN AROUND DOCUMENTS
- OBTAIN DATA FROM OTHER AUTOMATED SYSTEMS
- PROVIDE MACHINE READABLE DATA TO OTHER SYSTEMS

DATA ACCURACY AND COMPLETENESS

MEET CURRENT INFORMATION NEEDS

MEET CHANGING REPORTING NEEDS

INSTANTANEOUS ACCESS AND UPDATING

REGISTRANT CONSIDERATIONS

FIGURE 2

MOTOR VEHICLE SYSTEM PROCESSING CHART FUTURE ON-LINE PROCESS

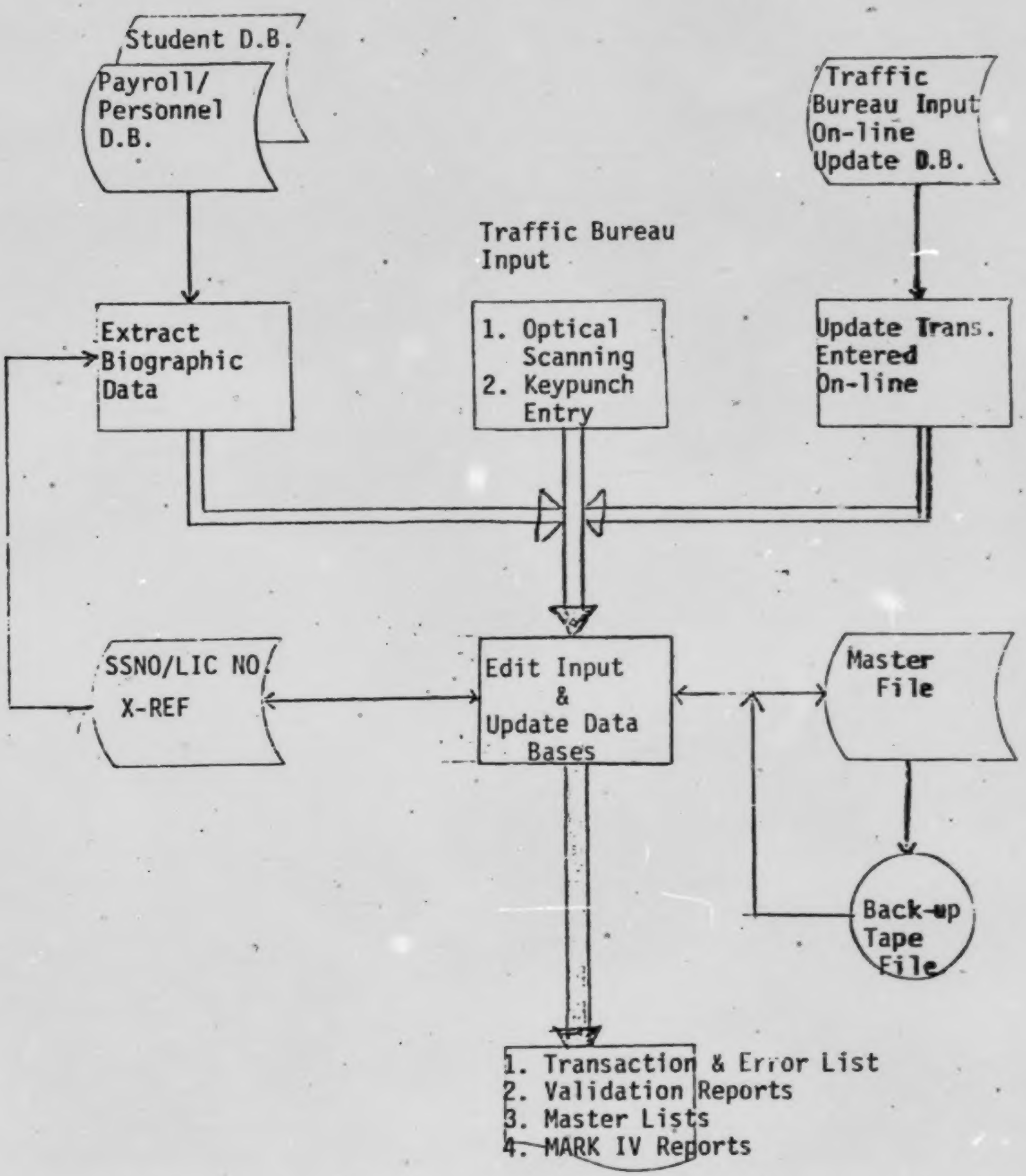


FIGURE 3

ON-LINE PROCESSING

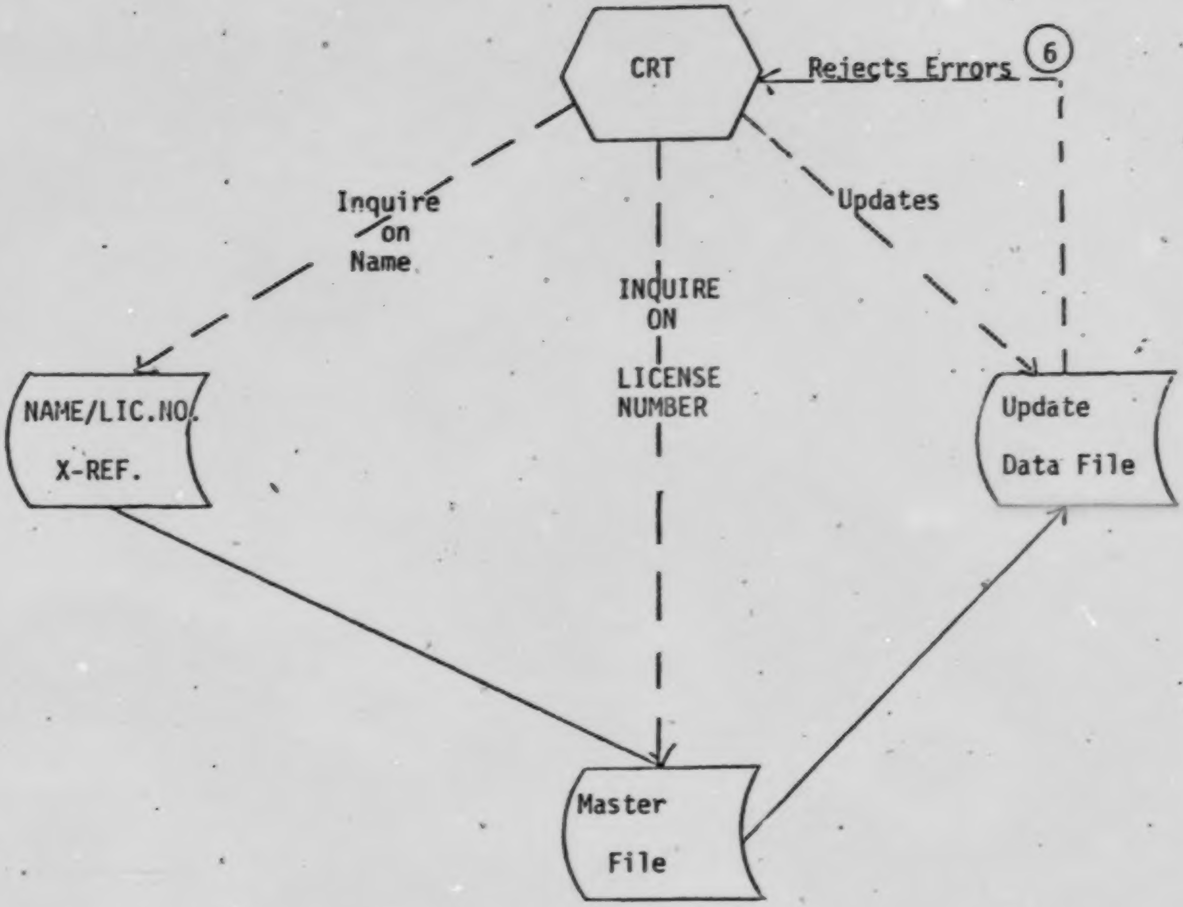
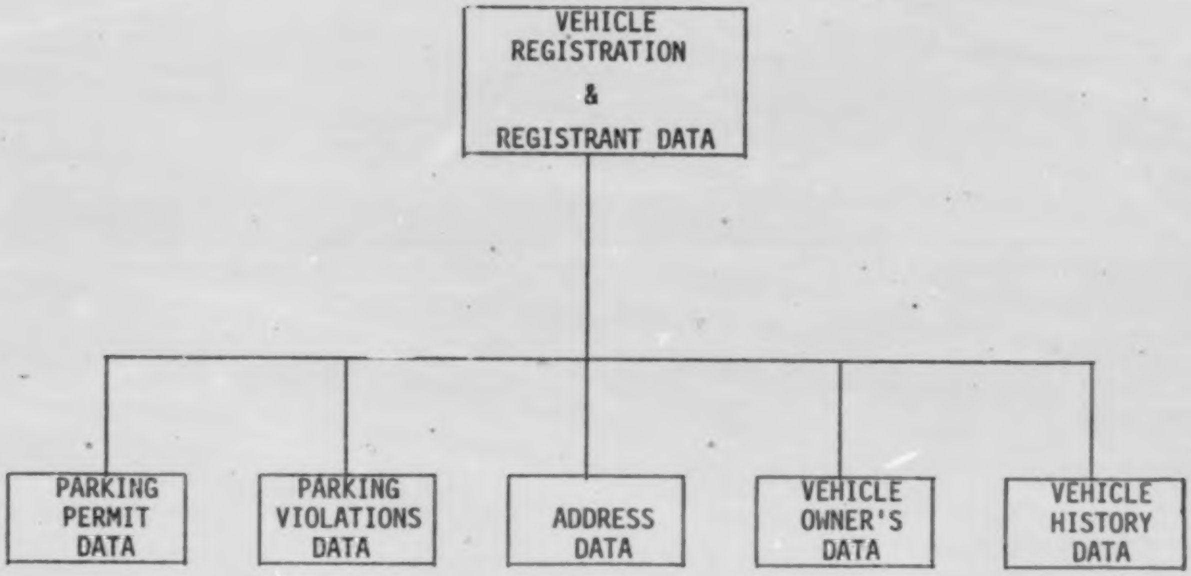


FIGURE 4

```
XX/XX/XX          MOTOR VEHICLE RECORD STATUS REPORT            PAGE XXX

LICENSE_NUM=XXXXXXXXXXXX  STATE=XX  REG_CODE=X  MAKE=XXXXX
YEAR=XX  BODY_TYPE=XXXX  COLOR_1=XXXXXX  COLOR_2=XXXXXX
ID_NUM=XXXXXXXXX  DEPT=XXXXX  TELEPHONE=AAAXXXNNNN  SPEC_ACT=X
U_PERMIT=X  N_U_PERMIT=X  LAST_CHANGE_DATE=YYMMDD
TERMINATION_DATE=YYMMDD  F_NAME=XXXXXXXXXXXXXXX
M_NAME=XXXXXXXXXXXXXXX  L_NAME=XXXXXXXXXXXXXXXXXXXXXX
SUFFIX=XXX  STUD_FLAG=X  P_CNT=XX  VIOL_CNT=XX  A_CNT=XX
C_CNT=XX  VEH_CNT=XX

1-PERMIT_TYPE; 2=PERMIT_NUM; 3=MULT_REG; 4=ISSUE_DATE;
5=VOID_DATE; 6=SPEC_REQ; 7=FEE_CODE; 8=REISSUE

   1=XXX  2=XXXX  3=X  4=YYMMDD  5=YYMMDD  6=X  7=X  8=X
   1=XXX  2=XXXX  3=X  4=YYMMDD  5=YYMMDD  6=X  7=X  8=X

1=VIOLATION_NUM; 2=VIOL_DATE; 3=DETAIL_CODE1; 4=DETAIL_CODE2;
5=OFFICER_NUM; 6=DISP_DATE; 7=DISPOSITION; 8=TOW_BOOT;
9=FINE_AMT; 10=ADJUST_CODE

   1=XXXXXX  2=YYMMDD  3=X  4=X  5=XXX  6=YYMMDD  7=X  8=X  9=XXXX.XX  10=X
   1=XXXXXX  2=YYMMDD  3=X  4=X  5=XXX  6=YYMMDD  7=X  8=X  9=XXXX.XX  10=X
   1=XXXXXX  2=YYMMDD  3=X  4=X  5=XXX  6=YYMMDD  7=X  8=X  9=XXXX.XX  10=X
   1=XXXXXX  2=YYMMDD  3=X  4=X  5=XXX  6=YYMMDD  7=X  8=X  9=XXXX.XX  10=X
   1=XXXXXX  2=YYMMDD  3=X  4=X  5=XXX  6=YYMMDD  7=X  8=X  9=XXXX.XX  10=X

ADDRESS_TYPE=X  COUNTRY_CODE=XXX  EXTRACT_CODE=X
ADDRESS=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  XXXXXXXXXXXXXXXXXXXX
        XXXXXXXXXXXXXXXXXXXX  XX  XXXXX

ADDRESS_TYPE=X  COUNTRY_CODE=XXX  EXTRACT_CODE=X
ADDRESS=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  XXXXXXXXXXXXXXXXXXXX
        XXXXXXXXXXXXXXXXXXXX  XX  XXXXX

1=OWNER'S_NAME;  2=RELATION_SHIP_CODE;  3=OWNER_NUM;

   1=XXXXXXXXXXXXXXXXXXXXXX X X  2=X  3=X
   1=XXXXXXXXXXXXXXXXXXXXXX X X  2=X  3=X

1=MAKE;  2=YEAR;  3=BODY_TYPE;  4=COLOR_1;  5=COLOR_2;
6=CHANGE_DATE;

   1=XXXXX  2=XX  3=XXXX  4=XXXXXX  5=XXXXXX  6=YYMMDD
   1=XXXXX  2=XX  3=XXXX  4=XXXXXX  5=XXXXXX  6=YYMMDD
   1=XXXXX  2=XX  3=XXXX  4=XXXXXX  5=XXXXXX  6=YYMMDD
```

FIGURE 5

MOTOR VEHICLE DATA BASE



PHYSICAL PLANT OPERATIONS
COST MANAGEMENT

Don Hennings
Systems Analyst
Administrative Data Processing
University of Northern Iowa

PHYSICAL PLANT OPERATIONS COST MANAGEMENT

D. Hennings
University of Northern Iowa

The maintenance of buildings and grounds is probably the third largest budgeted expense at a university; following faculty salaries and housing and food services. At the University of Northern Iowa, it is approximately 10% of our total budget. For this reason management felt it was essential to develop a system of cost analysis for plant operations. The system developed by UNI provides data for internal job costing, and data for management questions such as:

> For each of our labor crafts (electricans, plumbers, etc.) how is the time spend by type of work (preventative maintenance, modification, emergency repair) etc.?
>
> What are the total operational and maintenance costs for each of the University's vehicles and major pieces of equipment?
>
> What are the maintenance and utilities cost for each of our buildings; how is this cost distributed by type of work (preventative maintenance, emergency repair, etc.); how is this work distributed by building component (electrical system, roof, plumbing, windows and exterior, etc.)?

Answers to each of these questions, and others, provides data for management decisions on controlling maintenance costs. Examples of these decisions are revising or initiating a preventative maintenance program, replacing or rebuilding high maintenance cost equipment; modification or demolition of obsolete building structures, etc.

The UNI system requires

- job numbers be assigned to projects
- each job number include a cost center and operation number
- each craft have a code, and each worker report 100% of his time to a job number (including a job number for break periods, sick time, vacations. etc.)
- all materials whether purchased or from University stock be charged to a job number
- each job number have an account number for the internal charging of jobs within the University

The assignment of job numbers, cost center number, and operation number is essential to the system and is performed centrally within the Plant Operations Office. All of the work force report their time and materials to job numbers, the other information is added from the data processing master file. In this case study I will describe the number system, the inputs and outputs from Data Processing, and an outline of the system requirements. The system is in the CAUSE library.

.
.
.

In summary we have found this to be a straightforward procedure for reporting by our labor crafts, and the system provides basic information to assist decision-making by the Plant Operations management and University administration.

Editor's Note: Since the documentation for this system is in the CAUSE Library, exhibits and reports referred to in the excerpts, are not published with this paper.

NUMBER DESCRIPTIONS

1.0 JOB NUMBER

Special jobs will have a 4-digit number assigned by the Shops secretary. She keeps a log of these numbers and assigns them sequentially.

Standing and maintenance jobs will have a 5-digit number assigned that is made up of two parts. The first 3-digits will be the cost center number where the work is to be done. The second part will be the 2-digit operation number which identifies the work being done.

2.0 ACCOUNT NUMBER

Account number will be a 7-digit number that identifies the account that will be credited or debited with charges. These numbers are assigned by the Business Office.

3.0 COST CENTER NUMBER

The cost center number is a 3 character code that identifies an item or area where work is performed. Examples of a cost center number are a building number or a vehicle number.

4.0 OPERATION NUMBER

The operation number is a 2-digit number which identifies the general type of work being completed at a cost center. Operation numbers are assigned by the Physical Plant.

5.0 CRAFT NUMBER

Craft number is a 2-digit number which identifies the employee to a specific shop. Craft number will be assigned by the Physical Plant.

PROVIDING JOB ORDER INFORMATION TO DATA PROCESSING

1.0 Maintenance and standing job orders and all special job orders must be set up on Data Processing files. The information will be supplied to Data Processing on the JOB ORDER form (Exhibit A) or STANDING AND MAINTENANCE JOB ORDER SETUP form (Exhibit B) prepared by the Shops Office.

1.1 The information that Data Processing must obtain from the JOB ORDER and STANDING AND MAINTENANCE JOB ORDER SETUP forms is:

1.1.1 JOB NO. - Special jobs will have a 4-digit number assigned. A log of these numbers will be kept by the Shops secretary and are assigned sequentially.

Maintenance and standing jobs will have a 5-digit number made up of two parts. The first 3-digits will be the number of the cost center where the work is performed. The second part will be the operation number which identifies the type of work. To separate the two parts, use a dash.

1.1.2 DATE - This is the date when the job number was assigned.

1.1.3 DESCRIPTION - This should be a short title which would identify the job (limit to 50 positions including spaces between words).

1.1.4 ACCOUNT CHARGED - This should contain the account number that will be charged for the job. Following the account number should be the expense classification number, cost center number, and the operation number for the job. Separate each of the four numbers by dashes.

1.1.5 ESITMATED COST - Three cost estimates must be supplied. The first estimate must be the estimated material cost. The second estimate should be the estimated number of labor hours to do the job. The third estimate should be the estimated cost of labor.

1.2 Maintenance and standing job orders that exist at the end of a fiscal year will automatically be setup with zero material and zero labor charges for the coming fiscal year. Data Processing will provide Physical Plant with a list of these jobs at the end of each fiscal year. Jobs that no longer exist can be deleted by drawing a line through their description, return the list to Data Processing, and they will remove them from their files.

PROVIDING JOB ORDER INFORMATION TO DATA PROCESSING Cont'd

2.0 Data Processing will keypunch the information from the JOB ORDER or STANDING AND MAINTENANCE JOB ORDER SETUP and will update their files. The JOB ORDER or STANDING AND MAINTENANCE JOB ORDER SETUP forms will be returned to the Shops Office for filing.

3.0 Jobs that are closed must have the green JOB ORDER form sent to Data Processing. This form must contain the date the job was completed. Data Processing will keypunch this date and return the JOB ORDER form to the Shops Office. At the end of the fiscal year, all closed jobs will be removed from Data Processing's files.

PROCESSING COSTS CHARGED TO A JOB

1.0 There are three types of charges that will be made to a job; labor, direct material purchases, and materials transferred from the Shops' maintenance supply. For each of these charges, there must be certain information supplied to Data Processing so their files can be updated.

2.0 LABOR

Labor charges to a job will be obtained from the employees' JOB CARDS. For details see the write-up on JOB CARDS.

2.1 Labor corrections can be made by using the LABOR CORRECTION form (Exhibit C). This form will be used when reducing labor costs charged to a job and charging these costs to a Physical Plant account. The following information must be supplied on the form.

2.1.1 JOB NO. - must be the job number that is to be credited.

2.1.2 JOB NAME - is the title of the job.

2.1.3 PHYSICAL PLANT ACCOUNT NO. DEBITED - is the Physical Plant account number that will be charged with the labor correction and the expense classification for Physical Plant.

2.1.4 AMOUNT TO CREDIT JOB - is dollar amount that should be credited to the job.

2.2 LABOR CORRECTION forms will be sent to Data Processing weekly. They will keypunch the information and return the original form to the Physical Plant Office. Data Processing will print a report on all corrections for the week. This report should be checked in Physical Plant to make sure all labor corrections were processed. If there were errors, prepare LABOR CORRECTION forms to correct the errors and Data Processing can process these corrections at their next week's run.

3.0 MATERIAL TRANSFERS

Materials transferred from the Shops' maintenance supply will be charged to a job by using the MATERIAL TRANSFER form (Exhibit D). The following information must be supplied to Data Processing on this form.

3.1 DATE - should be the date when the material was used on the job, and should be in the numerical month, day, year format.

PROCESSING COSTS CHARGED TO A JOB Cont'd

3.2 ACCOUNT - (on left side of form) should be the job number that is to be charged for the material.

3.3 TOTAL - is the total cost of material listed on the form.

3.4 ACCOUNT - (on right side of form) is the account number that is to be credited with the TOTAL amount of the transferred material, and the expense classification number.

3.5 The MATERIAL TRANSFER form will go to Data Processing each week to be keypunched and will then be returned to the Shops Office for filing in job number order. An adding machine tape of all material transfers should be made and sent with the MATERIAL TRANSFER forms to Data Processing. Any errors detected when Data Processing updates their files must be cooperatively corrected by Data Processing and the Shops Office.

4.0 DIRECT MATERIAL PURCHASES

After a REQUISITION for direct material purchases (Exhibit E) has been written by the Shops Office, it will be sent to Physical Plant. Three typed copies will be made of the REQUISITION. Copy 1 will be sent to the Business Office for their use in preparing a purchase order. Copy 3 will be sent to the Shops Receiving area. The hand written copy will be filed in Physical Plant, alphabetically by vendor name. Copy 2 will be sent to Data Processing to provide them with the following information.

4.1 REQ. NO. - is the Physical Plant's requisition number for this requisition.

4.2 DATE - should be the date when the requisition is prepared in the Physical Plant and should be in the numerical month, day, year format.

4.3 VENDOR - is the vendor's name limited to 30 positions including spaces.

4.4 Typed on the back of the requisitions must be the breakdown of requisitioned material by jobs. If the requisition is for both Shop material and direct purchases, the breakdown must be in two parts to separate direct purchases from Shop inventory material. If the requisition is for Shop material only, the requisition should be sent to the Business Office, bypassing Data Processing.

4.4.1 Material on the requisition that is for direct purchases to specific jobs should be listed first. Heading this group should be the account number of the clearing account that will be charged for this material. The job number, items on the requisition that are for the job, and total amount for the job should be typed for each job on the

PROCESSING COSTS CHARGED TO A JOB Cont'd

requisition.

4.4.2 Material on the requisition that is for the Shops should be listed as a group by itself. The Physical Plant account number being charged, the items on the requisition being purchased and being charged to the account, and the dollar amount being charged to the account must be typed for Shop material requisitions.

4.5 The material requisition, copy 2, with the exception of those requisitions containing only items for Shop inventory, will be sent to Data Processing. Data Processing will keypunch the information they need and will then return the requisition to Physical Plant for filing in requisition number order.

5.0 As material is received in the Shops, invoices will be approved by Shop personnel. The invoice will be received in Physical Plant with the net amount indicated. Physical Plant will pull the requisition that matches the invoice. If the invoice contains only part of the material originally requisitioned,the items received will be checked and dated on the requisition. For all items that were charged directly to a job and are on the invoice, make an entry on the DIRECT MATERIAL CHARGE form (Exhibit F), one entry for each job. On this form write the job number, requisition number, vendor name, total dollar amount of the material invoiced that is charged to the job, and if this is the job's last item on the requisition that is to be received, write an 'F' in the column headed 'FINAL INVOICE'. The 'F' indicates that any encumbrance balance charged to the original requisition for the job will be zeroed out.

5.1 Each week Physical Plant will make an adding machine tape of all amounts on the DIRECT MATERIAL CHARGE forms for the week. These forms and the adding machine tape will be sent to Data Processing to be keypunched. The forms will be returned to Physical Plant after Data Processing checks the amount they processed to the amount on the adding machine tapes. If there is a discrepancy between the two amounts, the errors will cooperatively be corrected by Data Processing and Physical Plant.

JOB CARD

1.0 JOB CARDS will be used by Physical Plant employees to provide daily information about the work performed during the day, as well as time used for sick leave and personal time off.

1.1 JOB CARDS will be preprinted for each Physical Plant employee required to report his time for job costing. On the second Tuesday of July, October, January, and April, Data Processing will print a three month supply of JOB CARDS for each employee.

1.2 New employees that will require JOB CARDS must have the information needed to print the JOB CARDS supplied to Data Processing on the special form entitled JOB CARD INFORMATION form. Instructions on how to fill out this form are attached.

1.3 Employees who must have their craft number or Physical Plant account number changed on the preprinted JOB CARD will have this information supplied to Data Processing on the JOB CARD INFORMATION form by the Physical Plant.

1.4 For new employees (step 1.2) and employees whose craft number or or Physical Plant account number change (step 1.3), enough JOB CARDS will be preprinted to finish the current quarter.

1.5 If an employee does not have a preprinted JOB CARD, he can fill out a blank JOB CARD including name, social security number, craft number, and Physical Plant account number. A supply of blank JOB CARDS can be obtained from Data Processing.

2.0 Employees will turn in their JOB CARDS to the Physical Plant at the end of each day.

2.1 At the end of each week, the employees' JOB CARDS will be used in Physical Plant to provide hours information required for Physical Plant to fill out the employees' TIME CARD.

2.2 Overtime reported on the employees' JOB CARDS may be changed in situations where the employee does not work the required number of hours in the week to earn overtime. This change can be accomplished by crossing out the overtime hours and including them in regular hours.

3.0 After all JOB CARDS are in and the employees' TIME CARDS have been filled out, two adding machine tapes should be made of the number of hours being reported on the JOB CARDS. One tape should include regular hours and the second tape should include only overtime hours.

3.1 Each Tuesday, the adding machine tapes and the JOB CARDS will be sent to Data Processing where the JOB CARDS will be keypunched.

JOB CARD Cont'd

The keypunched information will be used to update Data Processing's job accounting files. The adding machine tapes will be used as a check to determine whether or not all of the JOB CARDS have been processed correctly. If errors exist, Data Processing and Physical Plant will cooperatively determine the source of the discrepancy and correct the error.

3.2 After Data Processing's job accounting files have been updated correctly, the adding machine tapes and the JOB CARDS will be returned to Physical Plant for permanent storage.

INSTRUCTIONS ON HOW TO FILL OUT THE JOB CARD INFORMATION FORM

The JOB CARD INFORMATION form (Exhibit G) is used to notify Data Processing of employees that must fill out JOB CARDS for job costing. Completed sheets as well as partially completed sheets will be sent to Data Processing each week, so that Data Processing will have the necessary information to preprint the JOB CARDS. This form is used to:

1. Include an employee so he will get a preprinted JOB CARD.
2. Change the information (craft no., or account no.) that is preprinted on the employees' JOB CARDS.
3. Delete an employee so that he will not get preprinted JOB CARDS.
4. Cause the employee to have extra JOB CARDS printed.

1.0 Social Security No. - must be filled out for each record on the sheet.

2.0 Name - last name, first name, and middle initial.

3.0 Craft Code - this is the two character code which identifies the employee's craft within Physical Plant.

4.0 Account No. - is the seven position account number of the Physical Plant where the employee is budgeted.

5.0 Add - if adding an employee to have JOB CARDS printed, write 'A' in this column; otherwise leave this column blank.
If an 'A' is coded, social security number, name, craft code and account number must also be provided for this employee.

6.0 Change - if changing an employee's craft code or account number, code 'C'; otherwise leave blank.

7.0 Delete - if an employee is to be removed from the group receiving preprinted JOB CARDS, code 'D' in this column; otherwise leave blank.

8.0 JOB CARDS Required - if an employee does not have enough preprinted JOB CARDS, it is possible to request extras by writing the required number of extras in this column. Valid numbers will be 1 through 99. If no extra copies are required, leave this column blank.

For employees that are checked as ADD's or CHANGE's, a supply of JOB CARDS to finish the current quarter will automatically be printed. If a different number of JOB CARDS is required for ADD's or CHANGE's, provide this number.

INSTRUCTIONS ON HOW TO FILL OUT AN EMPLOYEE'S JOB CARD

The JOB CARD (Exhibit H) is used by Physical Plant employees to indicate on what jobs they worked and for how long. Each employee that reports his time to a job will fill out a JOB CARD daily. One line should be filled out for each job worked on by the employee for that day.

1.0 NAME - Name should be preprinted; if not, write it at the top of the sheet.

2.0 SOCIAL SECURITY NO. - Social security number should be preprinted; if not, write the employee's social security number after his name.

3.0 CRAFT NO. - Craft number should be preprinted; if not, write the employee's craft number.

4.0 ACCOUNT NO. - Account number should be preprinted; if not, write the number of the Physical Plant account in which the employee is budgeted.

5.0 REGULAR HOURS - This column should contain the number of hours worked on the job, except for overtime hours.

6.0 OVERTIME HOURS - This column should contain the number of overtime hours worked on the job during the day. This number may have to be changed at the end of the week since the employee may not work the total number of hours required for overtime during the week.

7.0 JOB NUMBER - For jobs that have job orders prepared, this should be the job number on the job order form. If this is a maintenance job, use the 3-digit cost center number followed by the 2-digit operation number for the job number.

8.0 JOB LOCATION - Job location is a description of the geographic location where the work was performed.

9.0 DESCRIPTION OF WORK - Description of work should be a short description about the work that the employee performed.

REPORTS CREATED BY DATA PROCESSING

1.0 PHYSICAL PLANT CHARGES TO ACCOUNTS (Exhibits I & J).

1.1 These reports are printed monthly and are used to show the detail that was used to create the account charges the Physical Plant System passes to the UNI General Ledger System.

1.2 The reports are printed on 3-part paper. Copy 1 is forwarded to Physical Plant, copy 2 to the Shops, and copy 3 to the Business Office.

1.3 Three reports are created. There is a report for Direct Material Charges, Labor Charges, and Shops Material Transfers.

2.0 PHYSICAL PLANT CHARGES BY JOB (Exhibit K).

2.1 This report will be printed monthly.

2.2 Jobs that are indicated as closed during the month will print out on a separate report. One copy will be sent to the Shops Office and one copy will be sent to the Physical Plant Office.

2.3 All open jobs and all jobs that had activity during the month will be printed out on the second report. Copies will be sent to the Shops and Physical Plant Office and one copy will be sent to the department.

3.0 CHARGES BY COST CENTER, CRAFT AND OPERATION (Exhibits L, M & N).

3.1 These reports will be printed monthly and once each fiscal year.

3.2 One copy will go to the Shops Supervisor. Each foreman should get a copy of his craft. A third copy should go to Physical Plant.

PHYSICAL PLANT MAINTENANCE SCHEDULING

R. Wayne Ingersoll, Jr.
Administrative Systems Analyst
Connecticut College

PHYSICAL PLANT MAINTENANCE SCHEDULING

R. Wayne Ingersoll, Jr.
Administrative Systems Analyst
Connecticut College

The costs required to maintain buildings and equipment on college and university campuses is a significant portion of the budget. Connecticut College has developed a system for scheduling preventive maintenance of plants and equipment to prevent high cost breakdowns. This system was described in a February 1974 Technical Paper of the Association of Physical Plant Administrators of Universities and Colleges.*

---

* The Association of Physical Plant Administrators of Universities and Colleges, Technical Paper, Volume IV, Number 1, February 1974.

FEATURE ANALYSIS: PERSONNEL-PAYROLL SYSTEMS

David P. Tursan
Manager
Systems Development
University of Virginia

Feature Analysis of Personnel/Payroll Systems

David P. lursan
Manager, System Development
Administrative Computing Center
Office of Computing Activities
University of Virginia

The word payroll is defined in my dictionary as being first a list of employees, with the amount due to each and second the total sum of money to be paid out to employees at a given time. This definition is simple clear and understandable. One would expect payroll systems to time dependent listings of employees and monies. The automated (ie computerized) products entitled payroll systems most often are not simple, clear or understandable and do most everything but list employees and monies. How does this condition get created? In this authors simplistic viewpoint three causes can be uncovered:

1. A failure by the designers to recognize that Payroll and Personnel/Payroll systems are basic systems intended to provide record keeping functions on an operational level.
2. An opinion institution has of itself and its operating characteristics. Most institutions convince themselves that they are unique enough to obviate the importation and tailoring an operational payroll system in favor of tailor-made in house systems. i.e., I think that I am unique therefore I am.
3. A "date-stamp" set for the generation of computers in which the primary user representatives was matriculated. This date stamp, once established, tends to remain as a level of sophistication which is difficulr to alter and impossible to reset. Further since this user representative is the key element in the successful implementation of the system, this date-stamp sets the tone for type of system proposed.

Assuming that you accept the three causes or variations of them. Then the effects can be observed by evaluating standard personnel payroll systems implemented around the country. Most of them probably would not be imported to institution.

What then are the characteristics you have or should look for in a Personnel/ Payroll System? I have established five categories which can serve as a quality control checklist for any system. A collection of data:

1. Time worked in recognized units such as hours, percentage of effective time, or standard days.
2. Quantative information to satisfy governmental agencies for reports, taxes, and contributions. i.e. deductive information for individuals, 1099's, Blue Cross.
3. Qualitative information to satisfy current record keeping tasks of your institutions agencies - ie. Personnel or grant reporting offices.

B. Control of data

1. The provision for external and internal controls to audit and certify the renumeration of individuals by check, non-participating benefits, and accrual of benefits earned; this assumes the ability to reconcile all controls.

C. Compatibility of systems

1. The recognition and inclusion of linkage points for other institutional functions. In this category fall integrity to accounting and budgeting systems, WICHE-NCHEMS and HEGIS compatibility, multi-campus environments, and multi-faceted employed personnel. This personnel may or may not yet be hired -eg- split funded casual staff.

D. Reporting of information

The issuance of payroll checks and supporting documents also notices, lists, and reports to mirror historical information current activities, and expected events which will probably occur. These reports to be in human or machine-readable form as required. You can consider these needs as: Ad-hoc, time-dependent and time oriented needs.

E. Sophistication

1. Date orientation for all variable fields to effect a start/stop facility.
2. Multi-taxing authorities to anticipate the increasing needs for distributed taxation.
3. External table driven parameters to establish and monitor variable information. This includes tax rates, all deductions, and rate scales associated with positions.
4. Report extraction capability to interface appropriately with modules to print or transfer data internally.

To summarize the position we have covered to this point, the evaluation of your own current institutional information base is a pre-requisite for collecting, controlling and reporting information in a Personnel/Payroll System. Upward mobility to a more sophisticated or more inclusive system will be an effect of recognizing this information base.

The author has not addressed on-line or real-time requirements to this point due to its being a derived facility of importance to the university, directly related to the equipment installed in the institution, and an byproduct of the orientation of data processing towards data anda data based management systems.

Let us now review the systems registered in the CAUSE library. I have prepared a comparitive analysis sheet for the ten systems and will discuss the more important features of each.

SYSTEM CHARACTERISTICS

Institution	First Production Date	Machine Oper-Sys Language	Primary Orientation	Storage Media	Generation of System	Employee Types	Multiple System Interface	Positive or Exception	Deductions Allowed	Start/Stop in Effect	Consideration/Comments of the System
Private Large University	12/70	IBM360/40 DOS FORTRAN	Payroll	Disk	3rd	Hourly Salaried Student	No	Exception	100	No	I/O of old system was frozen, on-line reportings/ inquiry. Rebuild of an existing system, external table driven.
Public Large University	3/69	IBM370/155 OS/MFT COBOL	Payroll	Disk/ Tape	3rd	Monthly	Monthly Budget	Exception	100	Not needed	Possible on-line capability, good user orientation.
Community College	3/71	IBM360/25 DOS Assembly	Payroll	Disk	2nd/3rd	Hourly Salaried Contract	Accounting	Exception	99	No	No social security deductions, multi-schools allowed, normal & abnormal P/R.
Private Large University	10/71	IBM360/20 IBM360/65 OS/HASP PL/1	Payroll Personnel	Disk/ Tape	3rd	Hourly Salary	yes, as needed univ. acct. budget	Exception	20	Yes	Data Base oriented, well documented, table driven, no salary history.
Public Large University	1967	CDC3000 OS/OS FORTRAN	Academic Personnel Records	Disk	3rd	Academic Personnel	Budget	N/A	N/A	Yes	Good basic on-line personnel information system, monthly batch capability single input form.
Public Large University	1969	CDC3000 OS/OS FORTRAN	Non-Academic Personnel	Disk	3rd	Non-Academic	Budget	N/A	N/A	Yes	Good budget & position control. (See previous listing.)
Private Small University	12/71	Honeywell MOD 1 TR COBOL	Payroll	Tape	2nd	Hourly	Accounting System	Exception	8 salary charges 60 Deduct.	Yes	State & local tax routines not included, allows split department charges.
Multi-College	6/69	IBM360/40 DOS COBOL	Payroll/ Personnel	Tape	2nd/3rd	Hourly Salary Student	General Ledger System	Exception	20	No	Supports up to 9 colleges, good reporting capability.
Public Large University	7/73	IBM360/75 OS/MVT/HASP COBOL	Payroll	Tape	3rd/4th	Hourly	Budget	Positive	64	Yes	Master control file for all records, no paychecks produced, well documented, table driven.
Small Private College	9/70	UNIVAC 70-2 DOS COBOL	Student Payroll	Disk/ Tape	2nd/3rd	Student	Accounting Work Study	Positive	Taxes/FICA	Yes	Documentation not available to author as of December 1974.

USE OF ACT DATA AT KANSAS STATE UNIVERSITY

Mark E. Lipp
Applications Supervisor
Data Processing Center
Kansas State University

Founded in 1863 as the first land-grant institution established under the Morrill Act, Kansas State University has grown from a small agricultural college to a major university of six undergraduate colleges, a College of Veterinary Medicine, and the Graduate School. Areas of responsibility of the university include all state agricultural extension services, the State Agricultural Experiment Station with its five state-wide branches, and one of the few programs in the nation for milling and feed technology.

Enrollment in the university has been on a steady increase since its founding, with the Fall, 1974 enrollment being 16,422 (FTE 15,711).

The Office of Admissions and Records and the Data Processing Center both report to the Vice President for Academic Affairs, with most other users of data processing services reporting to the Vice Presidents for Business Affairs, Student Affairs, and University Development.

USE OF ACT DATA

Kansas State University receives data from ACT in two forms:

a. CFAR data - (Comprehensive Financial Aid Report) received weekly.

b. Test scores - received five times each testing year starting in November and ending in June.

USE OF ACT DATA AT KANSAS STATE UNIVERSITY

The CFAR data is received by the Office of Aids and Awards and is used for establishment of eligibility for receipt of financial aid.

The major user of the test score data is the Office of Admissions and Records for admission of applicants to the university.

OFFICE OF AIDS AND AWARDS

The Office of Aids and Awards is currently implementing a new system of record-keeping which will tie the CFAR data received from ACT with the admissions data received by Student Admissions, the ongoing student data maintained by Student Records, university and non-university scholarships handled by the university Endowment Office, federal, state and local grants, work-study, BEOG and GISL monies, loans through the NDSL system, and all other financial areas where establishment of financial need is required. This new system will be operational this year and will greatly relieve Aids and Awards of the massive amounts of paperwork required to administer these many different programs.

OFFICE OF ADMISSIONS AND RECORDS

The data on test scores received from ACT by the Office of Admissions and Records is used in three areas:

a. Automatic generation of letters of admission to Kansas high school graduates and certain non-Kansas students.

b. Determination of eligibility for admission for students not included in item a.

c. Generation of labels and listings for departments on campus based on the student's interest indications.

ADMISSION POLICY

The admission policy of Kansas State University is, to a large part, based on the location of the high school from which the applicant graduated. Since Kansas State University is a land-grant institution, all graduates from Kansas high schools must be admitted. Applicants that did not graduate from a Kansas high school are admitted based on their high school performance, with consideration given to their test scores from ACT.

Specifically, the following criteria are used:

a. Graduates from a Kansas high school - unconditionally admitted.

b. Graduates from non-Kansas high schools - must have at least a 'C' average in overall high school GPA, or be in the upper one-half of his graduating class and show promise on his ACT scores.

SUPPLEMENTAL APPLICATION FORMS

Through the process explained further on in this paper, certain students are admitted to Kansas State University automatically based on data received from ACT. This greatly simplifies the problems of the student in applying for admission since he is required to simply fill out missing information on a preprinted form and return it to the Office of Admissions and Records. This form, called the Supplemental Application Form (SAF, see Figure 1), is machine-printed for all students on the ACT files with the following criteria:

a. Senior in high school.

b. Kansas State University as first, second, or third choice.

c. Kansas high school graduate

OR

ACT composite score of 22 or higher and high school GPA of 2.5 or higher

d. Has not already applied for admission to Kansas State University.

The SAF is then mailed to the student. If the student returns the SAF, he is entered onto the Admissions Master File as an applicant.

AUTOMATIC GENERATION OF LETTERS OF ADMISSION (SAF)

Kansas State University receives from ACT a magnetic tape five times each testing year that contains data on all students in the previous testing period who indicated Kansas State University as a college choice. This data goes through the process indicated in Figure 2. The outputs of this process are:

a. Supplemental Application Forms (SAF, see Figure 1) for all students who meet certain criteria.

b. A listing (called the Non-response Listing) of all students for whom SAF's were printed.

c. A file of all data received from ACT so far in this testing year including this testing period.

d. A 'master' listing of all data so far received from ACT with an indication of students receiving SAF's.

The criteria for selection of students to receive SAF's are those mentioned in the preceding section of this paper. Students who have already applied for admission are eliminated by comparison of the ACT file with the Admissions Master File.

The Non-response Listing is used to check off those students who respond and also has been used to contact non-responding students to inquire as to why they have not returned the form.

When the student receives the SAF, he simply has to fill out any missing data (see Figure 1) and return it to Kansas State University to be admitted to school. The Office of Admissions and Records will then enter the updated information onto the Admissions Master File directly from the returned SAF.

OTHER USES OF ACT DATA ON CAMPUS

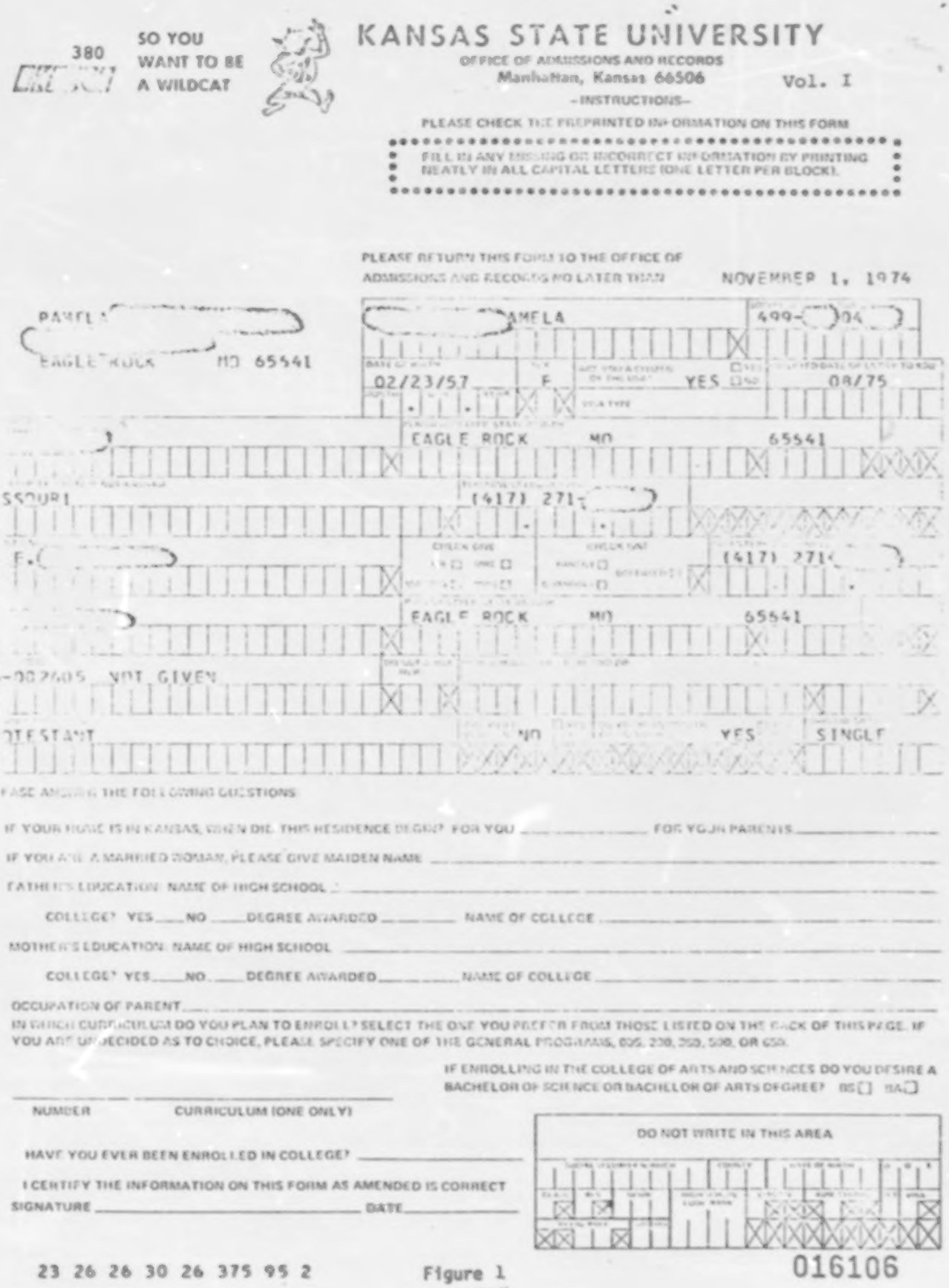
The use of the ACT data by other departments on campus has grown over the past several years. Currently, all involved departments receive an ACT Master List for each testing period that shows the students that have indicated an interest in a field of study in the departments' area. This helps the department in advising the student when he appears on campus and also gives the department an idea of the changes in interest areas of incoming students.

PROBLEMS ASSOCIATED WITH USE OF ACT DATA

Kansas State University has encountered only one problem in the use of the ACT test score data. The social security numbers as reported by the student to ACT are, many times, different from the numbers eventually received from the student by the Office of Admissions and Records. We conservatively estimate that ten to twenty percent of the numbers are changed as students go through the admission process. Since there is a matching process done on the ACT file and the Admissions Master File to eliminate printing SAF's for students having already made application, a hand matching process must be used as a follow-up to eliminate any social security number problems.

SUMMARY

Kansas State University is in its second year of using the ACT test scores for admission to the University. This has helped to improve the handling of applicants and has made it easier for the student to be admitted to the University. Few problems have been encountered in the process, and we plan to continue using the ACT data.



380

SO YOU
WANT TO BE
A WILDCAT

KANSAS STATE UNIVERSITY

OFFICE OF ADMISSIONS AND RECORDS
Manhattan, Kansas 66506

Vol. I

-INSTRUCTIONS-

PLEASE CHECK THE PREPRINTED INFORMATION ON THIS FORM

FILL IN ANY MISSING OR INCORRECT INFORMATION BY PRINTING NEATLY IN ALL CAPITAL LETTERS (ONE LETTER PER BLOCK).

PLEASE RETURN THIS FORM TO THE OFFICE OF ADMISSIONS AND RECORDS NO LATER THAN NOVEMBER 1, 1974

PAMELA [redacted]
[redacted]
EAGLE ROCK MO 65541

[redacted] AMELA

499-[redacted]04[redacted]

DATE OF BIRTH 02/23/57

F

YES ☐ YES ☐ NO

08/75

EAGLE ROCK MO 65541

[redacted]

ISSOURI

(417) 271-[redacted]

. F.[redacted]

CHECK ONE ☐ ☐ ☐ ☐

CHECK ONE ☐ ☐ ☐

(417) 271[redacted]

[redacted]

EAGLE ROCK MO 65541

-002605 NOT GIVEN

ROTESTANT NO YES SINGLE

PLEASE ANSWER THE FOLLOWING QUESTIONS

IF YOUR HOME IS IN KANSAS, WHEN DID THIS RESIDENCE BEGIN? FOR YOU __________ FOR YOUR PARENTS __________

IF YOU ARE A MARRIED WOMAN, PLEASE GIVE MAIDEN NAME __________

FATHER'S EDUCATION: NAME OF HIGH SCHOOL __________

COLLEGE? YES ___ NO ___ DEGREE AWARDED ______ NAME OF COLLEGE __________

MOTHER'S EDUCATION: NAME OF HIGH SCHOOL __________

COLLEGE? YES ___ NO ___ DEGREE AWARDED ______ NAME OF COLLEGE __________

OCCUPATION OF PARENT __________

IN WHICH CURRICULUM DO YOU PLAN TO ENROLL? SELECT THE ONE YOU PREFER FROM THOSE LISTED ON THE BACK OF THIS PAGE. IF YOU ARE UNDECIDED AS TO CHOICE, PLEASE SPECIFY ONE OF THE GENERAL PROGRAMS, 005, 200, 350, 500, OR 650.

IF ENROLLING IN THE COLLEGE OF ARTS AND SCIENCES DO YOU DESIRE A BACHELOR OF SCIENCE OR BACHELOR OF ARTS DEGREE? BS ☐ BA ☐

______ __________
NUMBER CURRICULUM (ONE ONLY)

HAVE YOU EVER BEEN ENROLLED IN COLLEGE? __________

I CERTIFY THE INFORMATION ON THIS FORM AS AMENDED IS CORRECT

SIGNATURE __________ DATE ______

DO NOT WRITE IN THIS AREA

23 26 26 30 26 375 95 2

016106

Figure 1

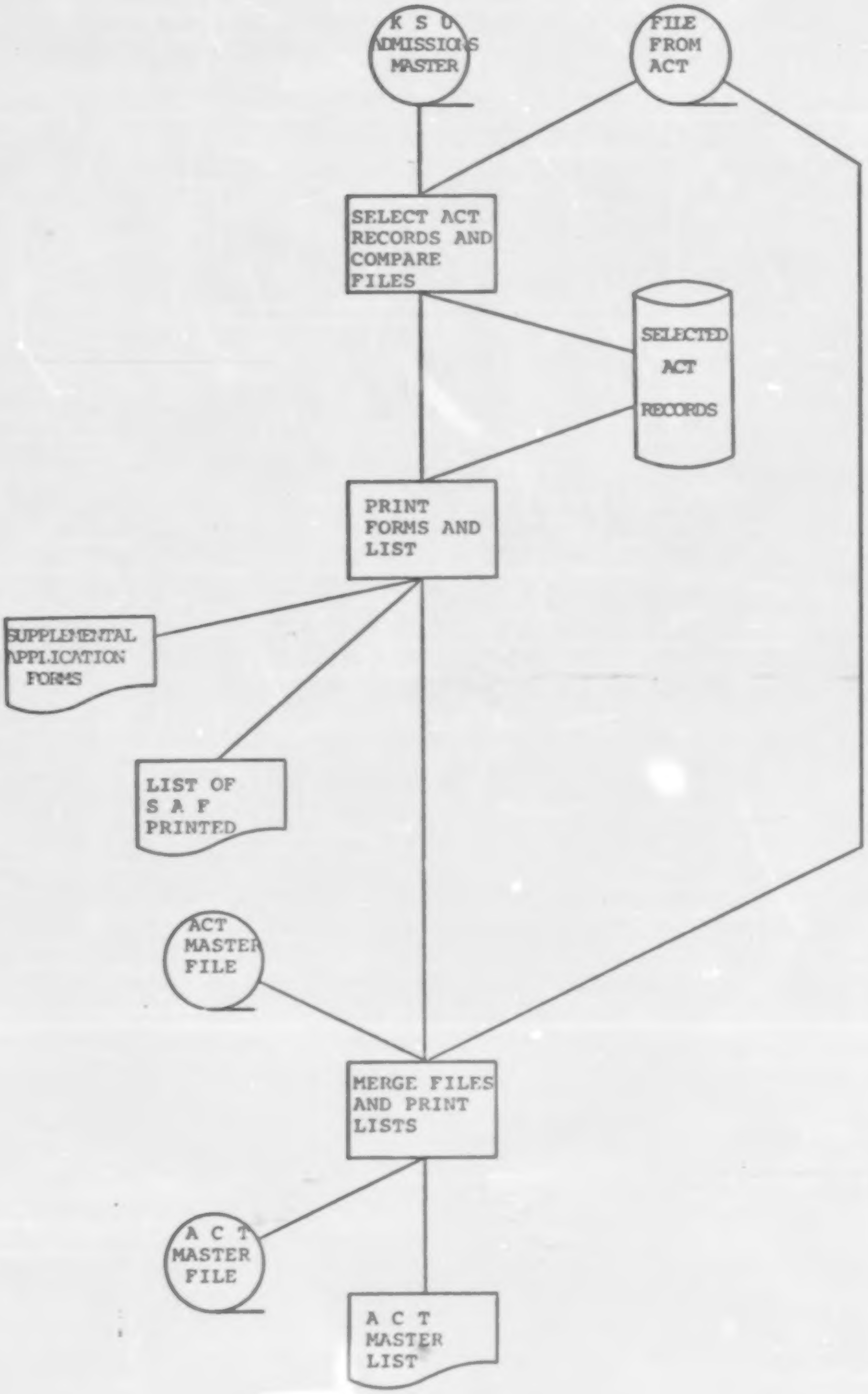

K S U ADMISSIONS MASTER
FILE FROM ACT
SELECT ACT RECORDS AND COMPARE FILES
SELECTED ACT RECORDS
PRINT FORMS AND LIST
SUPPLEMENTAL APPLICATION FORMS
LIST OF S A F PRINTED
ACT MASTER FILE
MERGE FILES AND PRINT LISTS
A C T MASTER FILE
A C T MASTER LIST

Figure 2

THE ROLE OF THE CONSULTANT IN SYSTEMS DESIGN

Peter Simmons
Booz, Allen, & Hamilton
Washington, D. C.

THE ROLE OF THE CONSULTANT IN SYSTEMS DESIGN

Peter Simmons
Booz, Allen, & Hamilton
Washington, D. C.

This presentation discussed the how and why of the use of consultants in higher education applications. Specifically, the following questions were addressed: How to select a consultant? When to select a consultant? And what are the range of activities for which a consultant should be used?

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

TRACK III

Data Base Systems

Tom Richley

Michael O'Connell

Adele Mildred Koss

Track Coordinator: Jane Ryland

WHY DATA BASE?

Tom Richley
Vice President
CINCOM Systems

Why is a New Look Required?

Since the late 1950's until the present, we have experienced an unbelievable technological evolution in automated data processing. We have evolved from crude mechanical equipment to computer systems. Within the computer systems evolution we have evolved from vacumn tubes, to transistors, to solid logic technology to chip technology. We have evolved from measuring performance in terms of a few instructions per second to millions of instructions per second. We have evolved from an environment where few organizations utilized computer systems to an environment where almost every commercial, governmental and educational organization has or is affected by computer systems.

It would be expected that the personnel in all areas of a modern organization, spending a significant portion of its budget for a large data processing department, should be involved in and pleased with the results of this technological revolution. Unfortunately, such is not the case. It is extremely disturbing that the non-data processing management within most organizations is disappointed with data processing. Such management evolved from initial lack of interest through excited enthusiasm to disenchantment.

At a time when data processing budgets have grown to such a size that high level management within an organization is turning its attention to the benefits returned to the organization for the expenditure, it has become extremely embarrassing for the data processing people to provide adequate justification.

The System of Management

Why is data processing not delivering better results to the organization? To answer this question it is necessary to review the history of the evolution of the "System of Management" within the organization and the evolution of data processing within the "System of Management."

What do we mean by the term "System of Management?" Every organization has a system of management typically structured as a hierarchy. A head man (President) manages a level of executives who manage levels of managers who manage workers. The structure of the system of management usually models functional areas necessary to perform the various decisions, functions and operations required to achieve the desired end results of the organization.

The Management Information System

In order for the organization to function, at all, there must be a system for processing and communicating information within and between the functional areas of the "System of Management." It is impossible for any "System of Management" to function without a "Management Information System." Today the term "Management Information System" is widely used. Many organizations say they are trying to implement a "Management Information System." Others say they have tried and failed. Others say that a "Management Information System" won't work for them. Others say they are going to implement a "Management Information System" in the future. All of these statements are ludicrous. Every organization has a "Management Information System." It has always had a "Management Information System." It could not exist without a "Management Information System."

What the above comments are really questioning is the contribution of the computerized data processing function within the organization to the "Management Information System." Since data is the basic raw material for a "Management Information System," it is quite disturbing that there should be a question as to the contribution of the data processing department to the "Management Information System" of the "System of Management."

The Evolution of the System of Management

To understand why data processing is not serving the "System of Management" and the "Management Information System" as it could, we must review the evolution of the "System of Management" and the "Management Information System" through three significant phases. To better understand this evolution let us use a hypothetical organization, Acme Toy Wagon Company.

The Era of Production

The Acme Toy Wagon Company was founded just after the depression. This era of economic rebirth, prior to World War II, could be called the Era of Production. Most growing organizations at this time were consumed with production. Usually a company could easily sell all that it could manufacture. The system of management of Acme Toy Wagon Company, at this time, was very simple. There was a President, a Manager of Production, a Manager of Engineering, a Manager of Administration and, perhaps, a small Marketing Department. In addition to the relatively small system of management, the product line was also simplified. They made small, medium and large red wagons.

World War II

The Acme Toy Wagon Company moved into the period of World War II. They suspended the manufacturing of toy wagons and began making rifle barrels, shell casings and trigger housings. Their main effort during this period was to increase production. As in the previous era, the "System of Management" remained simple.

The Era of Competition

As the end of the second World War approached, many economists feared a drastic economic slump. There was a concern whether the peace time economy could consume the output of the peak production capacity of World War II. Soon the Acme Company returned to the manufacturing of toy wagons. They were notified by one of the marketing representatives that one of their customers was purchasing wagons from another company who offered cigarette lighters as an optional feature to attract those who smoked young. No longer could Acme Company count on the fact that they could sell all they produced. This was the birth of the era of marketing, advertising and competition. Competition expressed itself in three ways:

(a) Feature competition - this competition centered around offering more varieties of and optional features for a particular product. Instead of just red wagons now they offered blue, green and two-tone wagons. In addition, they offered a number of optional features. One can see that the complexity of planning and scheduling products and procurring purchased items became significantly more complex as the the product line became more complex.

(b) Price competition - if a company could no longer secure or increase its share of the market through a variety of products and features, they were forced to compete by offering lower prices. To avoid the negative effect of lower prices, which is smaller profit margins, a company had to develop methods of reducing the costs. This need required more complex accounting, budgetory and planning controls.

(c) Delivery and service competition - in many cases, product differentiation or price competition became impossible. At this point companies in an industry began competition in offering increased services and shorter delivery time. In a standard product, in a standard cost and price environment, a company could compete by offering shorter lead times for delivery to their customers. Shortening of lead times often led to being required to forcast the customer demand because the lead time was less than the manufacturing cycle.

We can see the significance of the growth and complexity both in the "System of Management" and in the information required by that "System of Management", to cope with the complexities of doing business in this era of competition. As we have noted previously, the very simple "System of Management" of the Acme Company had now grown into a considerable number of divisions and departments performing specific functions and coming together as a whole to meet the objectives of the company. The importance of accurate information exchange between these numerous departments became critical to the performance of the company as a whole.

If we progress with the Acme Company to the later half of the 1950's, it might have been wise for the President to have called a meeting of the key executives and suggest the possibility that they should evaluate the method by which they were handling information with the hope that an improvement in this area could significantly improve the performance of the organization. Usually such a meeting was never called.

The Birth of Data Processing

Data Processing appeared in this system of management in some remote corner of the accounting department. Usually this embriotic new technology was applied to some simple clerical function such as printing payroll checks or summarizing sales history. This new department (Data Processing) appeared within the "System of Management" unnoticed by the organization. Any justification for automation had to be directly related to the replacement of clerical people and functions. At this time, an over-enthusiastic data processing department, in conjunction with vendors of data processing equipment, began to expand itself by reaching out and looking for more clerical functions that could be replaced to justify more data processing equipment.

Let's Automate Order Entry

In the interest of adding tape drives to a then small card oriented computer, the data processing department decided to persuade the order services department to replace the 15 girls who were manually typing orders by performing this function on the computer. After a long and reluctant period of persuasion, the manager of order services agreed to sacrifice the 15 clerks in exchange for computer prepared orders and the proposed benefits of better customer services, better control of inventory, and better decision making information. After considerably more elapsed time, cost, and human energy than promised, the customer orders were gloriously pouring forth from the computer's printer. To accomplish this feat, we had transferred filing cabinets of customer data to mysterious reels of magnetic tape. Similarly, we transferred ledger card files of inventory data to these magnetic storage devices. At this point, we expected the order services manager to be pleased with his move into the future. Unfortunately, he was not.

The Birth of the Report Syndrome

A month or two after the 15 clerks had been replaced by key punch operators and order input editing clerks and after the orders were pouring forth, the order services manager had the arrogance to indicate concern to the data processing people. He found himself void of information. His filing cabinets were empty. He required a daily orders received report by customer. A week later he required a daily orders received report by product. A month later he added the requirements for daily shipment reports by product and by customer. Shortly thereafter, he required a weekly inventory stock status and transaction register. The data processing people were delighted at this new enthusiasm expressed for their services. In the past, they had to plead for somebody to allow them to automate their data. Now requests for reports were pouring in. We could erroneously read this as success. Now, data processing became a requested and required function within the organization.

This pattern of development of data processing manifested itself in each department that ventured into the realm of automation. This set the precedent for the design of all data processing systems. First, convert data to tape files in such a manner as to satisfy the replacement of some particular clerical function or report and, secondly, to invent input documents to up-date these tape files and, thirdly, to sort these files and pour forth thousands of lines of information. This explosion in the volume of reporting lead us to the era of the "Operating System." During the 1960's, most of the hardware development, software development, and creative thinking were devoted to pumping more and more jobs simultaneously through a computer system. Little thought was given to the usefulness or quality of the information generated by these jobs. We never asked an end user why he required or how he intended to use the escalating lines of computer reports.

What If We Had Done It Differently?

As was stated previously, what if the President of Acme Company had called a meeting in the late 1950's and suggested to his key executives that he felt that the mishandling of information was causing many of their performance and cost problems? Perhaps he would have brought to their attention that forcasting the market for their products was, in itself, a difficult task but he was concerned that they were creating unnecessary "unknowns" by their lack of internal communication. Perhaps he suggested that they retain the services of a consulting firm to evaluate their information processing and to make recommendations to them. Let us assume that they did so and Mr. Jones was retained to conduct this investigation. Let us further assume that missed customer shipments was one of their biggest problems. Because of this, Mr. Jones decided to begin his investigation in the services department. Let us study his investigation and its conclusion.

Mr. Jones' first act was to introduce himself to Mr. Brown, the manager of Order Services Department. He asked Mr. Brown what his particular job function and duties were. Mr. Brown informed him that his department's mission was to process the orders for the company's products, to provide the paperwork to initiate shipment, to provide the necessary documents to the accounts receivable department to initiate billing and to gather evaluate and present information to production, specifying what and how much of specific products were to be manufactured for shipment or for storage in their warehouse. At this executive level, it was difficult for Mr. Jones to specifically detail the duties of Mr. Brown.

Mr. Jones then confronted a clerk named Mary and asked her what her duties and responsibilities were. Mary said that she was the Incoming Order Coordinator. Each day she gathered customer orders which had been received by mail, by teletype, by phone, or from salesmen. She sorted these orders into geographic areas and delivered them to specific clerks for each geographic area. Mr. Jones found it very easy to detail the exact duties of Mary. He noted also that she required no stored information to perform her duties nor did she make any decisions.

Next, Mr. Jones interviewed Jane. Jane processed orders for a particular geographic area. When asked to describe her duties and responsibilities Jane said that first she took her new orders to the customer file cabinets and withdrew the appropriate customer folders. Next, she took the orders to the inventory ledger card files, looked up the specific items, varified that enough of each specific item was available for shipment, noted the appropriate pricing information, subtracted the ordered amount from the item on the ledger card and recorded on the ledger card the assigned order number. Next Jane returned to her desk and, with the aid of a hand calculator and a typewriter, she computed the prices and typed the order. She prepared several copies - one copy was sent to the customer as an acknowledgement, one copy was sent to marketing, one copy was sent to the warehouse to authorize shipping and one copy was placed in the open customer order file. It was very easy for Mr. Jones to detail Jane's duties, responsibilities, and exact function. He noted that Jane required the use of stored information about the customer, the items, and the orders awaiting shipment. He noted, also, that her use of such information was threefold. Firstly, she extracted information from the customer and the inventory files; secondly, she up-dated information in the inventory file and; thirdly, she created and deposited information in the open customer orders file. Mr. Jones also noted that Jane made no decisions. In the course of further questioning, Mr. Jones asked Jane if she ever had any problems performing her function. Jane said, "Once in while I find that there is not enough of a particular item to meet the customer's shipment request." Mr. Jones said, "what do you do then?" Jane said, "I do nothing. I send a copy of the order to Mr. Smith. He handles back-orders."

In the course of his interview with Mr. Smith, Mr. Jones discovered a significant problem. Mr. Smith represented the first level of decision making management within the "System of Management". Mr. Jones discovered that it was difficult to obtain from this level of management specific facts as to how they perform their job duties and responsibilities. Mr. Smith informed Mr. Jones that it was very difficult to describe how he performed his job duties. Mr. Smith emphasized that many years of experience went into his decision making process. Mr. Jones recognized this and asked Mr. Smith if he might give him a general idea of his thinking process when he was presented with the problem of unavailable stock in inventory for a particular customer order. Mr. Smith said that he had to consider several factors and alternatives, summarized as follows:

(1) Are there any other orders for this item that have not been shipped as yet? Perhaps this current order could be given a higher priority.

(2) What, if any, current production orders for this item could be expedited to meet this customers shipment date?

(3) What is the sales volume to this customer? How important is he?

(4) What is the accounts receivable status of this customer? Does he pay his bills on time?

Mr. Smith said that after evaluating the above points, he selected the best course of action. If the customer could not be satisfied, he would discuss the matter with the marketing representative to the specific customer. Mr. Jones made a new discovery. Mr. Smith introduced a new method of utilizing information. Where Jane either referenced, up-dated or deposited information, Mr. Smith required the integration of information. Integration means the direct relationship between one logical center of information (inventory file) with another center of information (open customer order file). Mr. Jones also became disturbed about the quality of the information upon which Mr. Smith made decisions. He asked Mr. Smith, "How do you know the status of the open customer orders?" Mr. Smith responded, "We keep copies of all orders that we have sent to shipping in our files. The shipping department, at the warehouse, notifies us daily as to what orders they have shipped." Mr. Jones asked Mr. Smith, "How do you know the status of the production orders you have sent to the plant?" Mr. Smith replied, "We keep a record on the inventory ledger cards as to what production we have requested. The warehouse notifies us when they have received the requested production."

At this point, Mr. Jones left the Order Services Department and spent some time reviewing what he had learned. In the process of this review, he became concerned. The Order Services Department was located in the corporate headquarters office building. The warehouse was located some 15 miles from corporate headquarters and the plant was located in an adjacent town some 20 miles away. Mr. Jones jumped into his economy car and drove to the warehouse. He entered the office section and, to his amazement, there were a number of clerks busy working among many filing cabinets. One of those filing cabinets held requested shipments from the Order Services Department. Another set of filing cabinets contained ledger cards specifying the available inventory for specific products in the warehouse along with notations as to expected future production. Mr. Brown asked one of the supervisors how they prepared and reported shipment information to the Order Services Department. The supervisor told him that they kept a record of all shipments and sent the summary of the day's shipments to the Order Services Department each evening.

Quickly, Mr. Jones jumped into his economy car and drove the 20 miles to the plant. Upon entering the administrative office area of the plant, once again he noticed many clerks scurrying among many filing cabinets. Curiously, there was a set of filing cabinets which contained requested production orders from the warehouse.

At this point, Mr. Jones experienced a great revelation. He intended to file a formal report to the President, but he was so amazed at his discovery that he rushed immediately to the President's office. He informed the President that he had the solution to the problems. He had plotted the location of corporate headquarters, the warehouse and the plant on a map and formed a triangle by joining lines between them. He then picked a location equal distant from the points on the triangle and found that there was a few acreas of land available. He told the President, "What we have to do is build an information warehouse on this land and have corporate headquarters, the plant and the warehouse move all their filing cabinets into this location." He said, "The reason we are missing shipments and making inaccurate decisions is that it takes too much time for these departments to inform each other as to what they are doing. Mr. Smith, the order expeditor, is often making decisions to re-arrange priorities for customer orders that already have been shipped. He is also allocating expected production that may have been completed, significantly delayed, or cancelled without his knowledge. We should equipt our clerks with tennis shoes and have them sprint regularly between their offices and this central warehouse of data. Most of our problems in handling information center around the fact that we have several different versions of the truth. I found three different inventory files - none of which were in agreement with the others. We are experiencing too long a delay from the time that a business event occurs until it is reflected in the various files of information."

In reality, we know that Mr. Jones' recommendation to build a central warehouse for data and to require speedy clerks with tennis shoes is ludicrous. In reality, we know that Mr. Jones would not have made such a recommendation. However, this analogy is extremely revealing. What Mr. Jones might have said is, "Perhaps we can approach this new technology called 'computers' and find our answer." He might have recommended to the President that he contact a leading computer manufacturer and see if they can solve the problem. If the President had done so, he would have been disappointed because in the late 1950's the technology in computer hardware and software did not yet exist in a form that could be used to build this "central electronic warehouse for the storage of data." A wise decision on the part of the President would have been to indicate his requirement for the development of the necessary technology and indicate that he would wait anxiously for the technology to become available. Unfortunately, he did not wait and, unknown to him, data processing was born and matured trying to utilize technology which could only "sort and report."

Today many organizations are analyzing their large data processing budgets, realizing that they have invested great quantities of time and money and are far away from an electronic warehouse of data. If the technology had been available and if Mr. Jones had received additional consulting contracts to guide the implementation of a central warehouse of data, he would have realized that replacing the 15 clerks who were typing and processing customer orders was probably the least beneficial thing they could have done. Improving the accuracy and speed of applying the results of business events to the information centers (files) required by the organization and the consolidation of multiple versions of the files into one central "Data Base" would have provided the organization with a value many times in excess of the costs incurred. When we automated our clerical functions in the late 1950's, we did not realize that we were creating more problems for decision making management than we were gaining in reduced clerical costs. When the Order Services Department requested the endless reports from data processing we should have asked them why they required the reports. Their answer may have been enlightening. They would have informed us that we took all of their data away from them and they no longer had the ability to walk to a ledger card file and immediately determine the relationship of a particular item to the number of orders that had not yet been shipped. In order to get such information, the data processing people designed thousands of lines of reported information in various sequences from which they could interpulate their answers.

Today the software and the hardware exists which can enable us to implement a central warehouse of data. We call this new technology "Data Base". Data Base means developing a logical model of the "System of Information Centers" and the relationships between them necessary to model an organization's "System of Management" and the interaction between this "System of Management" required as business events occur.

A DATA BASE MANAGEMENT SYSTEM APPROACH

Adele Mildred Koss
Senior Systems Analyst
Harvard University

About 2 1/2 years ago the administration at Harvard decided to review and evaluate administrative data processing. Since each budgetary unit had designed and implemented its own system, it was soon evident there had been no attempt to develop a uniform approach to computer processing, or to determine what information was available from systems in other departments. The university was thus faced with a spectrum of unintegrated computer systems, and frequently could not get the information that was required both by the central administration and the Federal Government. This confrontation of almost a Tower of Babel was coincident with the rising interest in Data Base Management Systems. It seemed reasonable that a Data Base Management System could be used by the Administration as a vehicle for providing more unified and therefore more useable computer based systems. Individual departments could continue to initiate and develop their own requirements for computer systems, but the work would be done in conjunction with the Applications Development Group, within the Office of Information Technology. This group would review paper flow, analyze data requirements, reports, and develop system specifications with the funding department, keeping in mind a goal of consolidation of Harvard's processing as well as the individual department's needs. The same Applications Development Group had the charter to examine available Data Base Management Systems and select one which would be useful in Harvard's Administrative processing environment.

For the next year we examined and evaluated generalized Data Base Management Systems. There were really not as many working systems to choose from as the literature seemed to indicate. For all the publicity, the "state of the art" was not as sophisticated or advanced as it purported to be. The online systems, in particular, seemed to be utilized mainly as a test tool and in general little existed in the way of run statistics and cost analysis. The name Data Base Management System also seemed to be applied to a wide variety of package types which included access methods, retrieval methods, report writers, dictionary processing systems and various combinations thereof.

While our general goal was to centralize and, where possible, consolidate our programming efforts we were hoping for other advantages such as:

Reduced cost of new program development and program maintenance, since first and foremost a Data Base Management System should allow the user to access data in a logical way, independent of its physical location. This separation of I-0 processing should simplify programming.

Easier access to data hopefully leading to increased sharing of data and more timely availability of information.

Finally, an opportunity to evaluate on-line processing.

More specifically we were looking for a system which would provide all or some of the following attributes:

1. Logical data access, independent of physical placement of data
2. Data description language
3. Data manipulation language which could be used within a host language such as COBOL
4. Direct and queued processing
5. Multiple search keys
6. Security
7. User language interface for non-programmers, to be useable in on-line processing or in batch processing
8. Existing on-line system
9. Backup

It also seemed highly desireable to attempt to find all these capabilities in a single integrated package, that could function in a reasonably cost effective way, rather than a system of packages.

We evaluated several generalized Data Base Management Systems, visited sites where such systems were in operation, and ran tests at Harvard for more detailed analysis in our own environment.

These tests were conducted using the MIT Information Processing Center's IBM 370/165,which has subsequently been replaced by an IBM 360/168, this facility now has the following configuration:

3 megabytes of core

7 9-track tape drives

2 7-track tape drives

7-2314 disk drives which are available for private mountable disks.

12-3330 disk drives only useable for on-line disk storage.

The computer test runs provided comparative statistics on CPU [illegible] number of disk accesses, core requirements, core occupancy, and disk occupancy as well as indicated the ease of implementation. We decided on a package developed by Computer Corporation of America, Model 204 Data Base Management System which includes IFAM, an inverted file access method for procedural language data manipulation. It provided most of the capabilities we were anxious to test; its core requirement of 120K to 150K seemed minimal for an integrated package; its performance costs in testing were good in comparison to other systems; and the rental price was competitive.

The initial implementation plans were to develop on Employee Information Data Base for all employees at Harvard, and a Name and Address Base for all individuals, students, and employees, associated with Harvard. We chose modest goals and deliberately did not combine these bases although there is some duplication of data. Each base in itself was projected to be fairly large, and intial indications were that run costs would be high and somewhat proportional to the size of the base. We also wanted to proceed in a fashion that would allow us to evaluate individual system performance and its general impact on processing in the relevant departments. We did, however, coordinate our efforts and make certain the two bases would be compatible though developed for different budgetary units.

The remainder of this paper primarily describes our experiences in implementing the Employee Information System, but many of our conclusions were inforced by the Name and Address System.

The Employee Information System includes data for approximately 17,000 people. There are two types of records: one containing personal data, the other containing job data. Each individual has one personal data record and can have multiple job records. The personal data record may have up to 177 fields of data, and the job record may have up to 197 fields of data. Thirty-three fields have been designated as keys. The selection of key fields was made very carefully, since we learned in testing that the cost of maintaining many keys was expensive. A balance must be struck between retrieval savings & update costs. There are now over 40,000 records in the system and the data and keys already fill more than one 2314 disk pack.

The base implementation was accomplished in one year and required about 1 1/2 man years of programmer analysts' time. The system was to be user-transaction-oriented, and was actually a consolidation of two previous system developments. The specific tasks included in the effort were:

Design of uniform specifications and transactions for processing all Harvard employee data.

Design and implementation of the data base fields and the keys for searching.

Conversion of two existing Personnel Systems files to provide the first data base.

Development of an Edit program to be used to validate all data and check for transaction requirements before updating the data base.

Development of an update program which accepts the validated user-transactions and executes the required data base calls.

Development of report programs to provide Microfiche copies of the Employee Information Data Base.

Finally all other required reports were to be done in either the Data Base Management System User Language or by "File Exec" which is a retrieval and report writing system we have been using at Harvard to satisfy many reporting requirements. The computer costs were about $8,000 for check-out, and $11,000 for conversion and parallel running. Some of the initial high costs incurred in parallel running were reduced as we learned how to "fine tune" the system and to submit runs less expensively within Harvard's charging algorithm. There are other run costs and characteristics which also deserve to be mentioned.

1. The cost of direct access updating, with a data base with a great multiplicity of keys is extremely high. The high cost is due to the large number of disk accesses, the processing required to update keys, and the length of time core is occupied to perform these operations.
2. Deferred processing, updating the data directly but accumulating the key updates for a separate sequential pass, greatly improves run cost since total disk accesses are drastically reduced.
3. The CPU costs with this system were significantly higher than traditional access methods. Partially this is due to maintaining keys but it also seems to be a function of the compacting and expanding of fields to conserve disk space, and the interpretation of I-O calls required to access data logically in the base.
4. The economic advantages of random access for update are somewhat lost when the cost of frequent sequential back-up runs are included.
5. Complex application programs written with a host language, requiring a large amount of core, are very expensive to run, since the Data Base Management System itself requires 125-150K thus resulting in a very large combined core requirement. The further combined effect of large core requirement and lengthy core occupancy because of ex-

tensive disk accessing is bound to result in costly runs.

6. On-line processing, except in a very small base, was also not feasable in our computing environment. Since the system could only be brought up under TSO (an MIT requirement), the entire base would have to reside in on-line disk space, charged at $.60 per track per month. Alternately the Data Base could have been loaded and unloaded once each day at a cost of $60 a day. However we did try such processing, and feel there could be real advantage to on-line alteration of certain segments of data, and on-line preparation of data.
7. It is relatively simple to add new fields to the base but this would have greater advantage in a situation where many users are placing data in the same base. However in our situation, if the Employee Information Base held other data which required more disks to be mounted, and used more of the system resources, the running costs would be prohibitive.
8. In certain situations the user language interpretive compiler produces more efficient code than equivalent IFAM calls in a host language. However the user language is not powerful enough to perform all the required program functions.
9. The retrieval costs with a Data Base Management System such as Model 204 are most impressive when the search criteria are all keys and the resultant selection is reasonably small. Since the employee data exists in both data base form and sequential file form we were able to do the same retrievals using both the Data Base Management-User Language, and File Exec Language which is a sequential retrieval and extraction program. In both systems the requests are fairly simple to specify. Following are some charts based on comparable searches and extraction of data.

10. Obtaining comparative statistics on updating costs is not quite so simple. The best we could do is compile some update costs on the two previous personnel systems and display them along with current Employee Information Data Base Update Costs.

    The Wage and Salary System was a sequential file system with personal and job data for about 10,000 people - about 10 million bytes of data.

    The Corporation Appointment Record System utilized a sequential file together with an ISAM file. It contained personal data for about 10,000 people and job data for about 35,000 jobs altogether about 15 and a half million bytes of data.

    The Employee Information System is a consolidation of the two systems. It has fewer total records, and many similar user transactions but it also has some added capabilities. The base contains about 20 million bytes of compressed data, exclusive of keys. There is personal data for about 17,000 people and job data for about 24,000 jobs.

While there is a decided advantage to having all the data in one base and in uniform format, the running cost is clearly quite high. Our analysis indicates that much of the high costs is due to the processing required by the Data Base Management System.

As stated previously the experience with the Name and Address file corroborates our findings with Personnel, i.e.,update is expensive, retrieval with keys and small selection are fairly inexpensive. We currently tend to use the Name and Address Base largely for extracting data and to update other files but we do intend to try accessing the Employee Information Base and Name and Address Base together. In both systems we have found that by centralizing the information a significant improvement in the accuracy of the data has been obtained.

Our conclusions from these efforts tend to be, that coordinating the data and the development of systems can have real payoff. We have begun to maintain and acquire better information on the employees at Harvard. There has been some reduction in manual efforts in the Personnel department. It seems, however, that most of the benefits have come from consolidation of processing efforts, which certainly results as much from administrative decisions as from the advantages of data base management. Other uniform programming standards and access methods could provide the same benefits particularly if they have a capability for easy retrieval and reporting. We certainly have seen systems that seem more cost effective as the next paper will illustrate.

To date all our testing with data base packages has indicated that data base processing in its present state, with great reliance on random access, is quite expensive in a large computing center environment. It cannot compete in cost with sequential processing. Perhaps in a dedicated Administrative Computer Environment it would be economic to consolidate data into a single base and use the data and on-line terminals effectively. Clearly Data Base Management Systems do offer flexibility and opportunity for relating great quantities of data, but cost considerations cannot be ignored or underestimated. Furthermore the consolidation of systems and data is really an administrative decision independent of,although facilitated by data base techniques.

At the present time, we are investigating the utilization of mini-computers and their possible use in a distributed data base environment. Perhaps this will provide the flexibility and cost effectiveness we are seeking.

SYSTEM	NO OF KEYS	NO OF SEQUENT TEST	NO OF RECORDS SELECTED	NO OF RECORDS SEARCH	NO OF FIELDS EXTRACTED	CPU TIME MIN	CORE	OCCUPANCY MIN	DISK I-O	TAPE I-O	TOTAL COST
M204	5	0	500	0	6 FIELDS 67 BYTES	.046	230K	.354	599	6	$5.00
FILE EXEC	0	5	500	10,000	6 FIELDS 67 BYTES	.061	170K	.716	71	1,485	7.50

RETRIEVAL 1

SELECT RECORDS ACCORDING TO 5 KEYS AND EXTRACT 6 FIELDS FROM THE SELECTED RECORDS FOR OUTPUT.

SYSTEM	NO OF KEYS	NO OF SEQUENT TES	NO OF RECORDS SELECTED	NO OF RECORD SEARCH-ED SEQ	NO OF FIELDS EXTRACTED	CPU TIME MIN	CORE	OCCUP-ANCY MIN	DISK I-O	TAPE I-O	TOTAL COST
M204	2	2	200	14,000	6 FIELDS 60 BYTES	.903	200K	2.680	3530	3	$31.00
FILE EXEC	0	4	200	16,000	6 FIELDS 60 BYTES	.132	150K	1.117	70	2,278	$12.00

RETRIEVAL 2

1. SELECT A SET OF RECORDS (SET A) ACCORDING TO TWO KEYS.
2. TEST TWO FIELDS IN SET A TO SELECT A SUBSET OF RECORDS (SET B)
3. EXTRACT 6 FIELDS FROM RECORDS SELECTED IN SET B FOR OUTPUT.

SYSTEM	NO OF KEYS	NO OF SEQMENT TEST	NO OF RECORDS SELECTED	NO OF RECORDS SEARCHED SEQUENT	NO OF FIELDS EXTRACTED	CPU TIME MIN	CORE	OCCUNPANCY MIN	DISK I-O	TAPE I-O	TOTAL COST
M204 No LIST COMPARE	5	1	1699	16,000	6	4.244	230K	72.25	135,982	17	$620.
USING LIST COMPARE	5	1	1699	16,000	6	2.321	230K	26.51	48,346	18	$236.
FILE EXEC	0	6	1699	40,000	6	.163	120K	2.30 MIN	83	5,028	$ 20.

RETRIEVAL 3

1. SELECT SET OF PERSONAL RECORDS (SET A) USING 2 KEYS-EXTRACT 2 FIELDS OF PERSONAL DATA.
2. SELECT SET OF JOB RECORDS (SET B) USING EXTRACTED FIELDS AS ONE KEY PLUS OTHER KEY CRITERIA.
3. EXTRACT JOB DATA AND CONCATENATE WITH PERSONAL DATA FOR OUTPUT.

WAGE AND SALARY 10,000,000 BYTES SEQUENTIAL FILE ON TAPE 10,000 PEOPLE & JOB DATA		CORPORATION APPOINTMENT RECORD SYSTEM 15,625,000 BYTES SEQUENTIAL FILE ON TAPE PARALLEL ISAM FILE ON DISK 10,000 PEOPLE & 35,000 JOBS		EMPLOYEE INFORMATION SYSTEM 20,000,000 BYTES OF COMPRESSED DATA IN DATA BASE 17,000 PEOPLE 24,000 JOBS	
#TRANS	COST	#TRANS	COST	#TRANS	COST
307	$19.27	79	$19.25	93 (ONE FIELD) CHANGE	$10.59
809	$21.16	210.	$18.13	148	$48.39
981	$24.76	490	$49.03	809	$130.39
1024	$22.84	524	$33.05	911	$168.99
3896	$35.24	1717	$49.86	1090	$304.78
		4892	$75.13	3600	$509.46

UPDATES

A MORE CONVENTIONAL SYSTEM APPROACH

Eugene E. Ekstrand
Senior Systems Analyst
Harvard University

A MORE CONVENTIONAL APPROACH
OR
A VOICE FROM THE DARK AGES

About a year ago C. W. Bachman[1] in an article in the Communications of the ACM said that those programmers who work with sequential files are still living in a Ptolemaic world while those who have advanced to the use of the Data Base concept are in the Copernican world. In fact, Bachman won't even call a sequential file a data base. So I am here to talk about non-data base processing. And, since the programming was done in ADPAC, the non-data base is processed with a non-language. That is, according to Jean Sammet's[2] book on programming languages. She rules out anything like Autocoder, RPG, or decision tables, which must be written on a form. This all reminds me of "Through the Looking Glass" - "When I use a Word" Humpty Dumpty said, in a rather scornful tone, "It means just what I choose it to mean - neither more nor less".

Another handicap for our project, in addition to our dark ages approach, was that we did not start with a charter to analyze the information paths of an organization, identify the sources and sinks of information, develop a model, design a data base, etc, etc,. We started in 1970 with the remains of an unsuccessful attempt by a software consulting firm to create a system for Harvard University. The decision was made to try to make use of the master file that they had created, although their update program aborted every time it was run. About 150,000 transaction cards had been coded and keypunched and never used. The master file contained about 200,000 records. Our plan was to transform the master file into a very simple format, make use of the transaction cards, and allow the clerical staff to work on cleaning up the data. At the same time another project group would experiment with IBM's Generalized Information System and ultimately we would create a GIS Data Base from the simple master file. Unfortunely the GIS experience indicated that the running cost would be more than unreasonably high. So that plan had to be dropped. We were then in a situation where we could update the master file, had used the transactions, and could perform retrievals. For inqueries we were using File Exec, a very inexpensive general query program, inefficient but easy to use. But this somewhat unplanned, supposedly interim system, had some major drawbacks. We had a sequential file of 200,000 records, each record was 1,116 bytes in length, almost a quarter of a billion bytes. The file occupied 10 reels of tape at 800 BPI and maximum blocking. The update run occupied our IBM 360/65 for 3 to 4 hours and cost about $100.00. On IBM 2314 disks, the file would have occupied 8 units. Many of the fields in any given record were blank, since we had kept the record structure very simple to allow for easy conversion to a generalized Data Management System.

At that point we were processing a fairly large amount of data, but rather slowly and inefficiently. But - at least the programs worked dependably, a pleasing and unusual experience for the users who had suffered through the 2 earlier attempts. But, we still had to pay $100. to $300. for retrievals and $100. for updates. The design of the master file and the transactions were not good since our goal had been - keep everything very simple to make conversion easy. Now our problem was to make our processing more economical, without a major conversion project, and without buying a high priced general package.

The solution to this problem turned out to be fairly simple. First two algorithms were developed to compress and expand the data. Any string of 4 or more blank characters is discarded and replaced with a string terminating character and a 2 byte binary space counter. This reduced the size of our master file to 3 reels of tape from the original 10. It also lowered the price of updates from $100. to $50. and reduced machine occupancy from 3 or 4 hours to 30 minutes. The compacted records are expanded only when hit by change transactions. We have since gone to 1600 BPI tapes so our master file now fills 1 and 1/3 reels.

This step, while fine for updating, would have had the effect of increasing the cost of retrievals. Therefore, we developed another condensed index file of short fixed length records. The file contains the most frequently used key values in the master file and in some cases an indicator that repetitive fields in the master do contain some significant value. These fields are automatically updated by transactions generated in the main update program. The file also contains a list of codes which do not exist in the master. These are used by the Data Base Manager for a variety of reasons that are unknown to the programming staff. He can introduce new users to the system by assigning one or more of these codes for their use, identifying some sub-set of the file for their special use. The responsibility for maintaining this field then belongs exclusively to that user and no systems or programming work is required at the computing center.

The data being processed is relevant to the Alumni Records Office, to the University Development Office, and to the Mail Communications Center, who in turn are responsive to 40 or 50 using offices. Our contact with all of these people is through William W. Mercer who, among other things, performs the duties of a Data Base Administrator. A large proportion of the processing is for the creation of mailing labels for various sub-sets of the file. The file will also be used for the printing of the Alumni Directory. We also print selected sub-sets in a variety of formats including microfiche. Each record may contain up to 1,116 bytes organized in up to 5 levels of hierarchy. There are 190 elementary data items and 31 highest level data groups. Some of the fields are repetitive existing in from 3 to 10 segments.

Inquiries first scan this file, retrieving a list of keys and attaching a retrieval batch number. This list is then passed against the master file and only the required records are expanded. The records are expanded only to the point where the requested output data exists. We now run retrievals for a cost of 1/10 of a cent to 5 cents per retrieved record. Since the nature of the data usage does not require instant access, overnight turnaround is adequate. We can, when required, perform retrievals in 1 or 2 hours.

The number of output records generally ranges from 5,000 to 30,000, but has been as low as 5 records and as high as 170,000. The worst case, as far as cost goes, was the retrieval of 5 records from the total of 200,000. The cost per record was about $4.00, apparently the requestor needed a quick response to a question that could not easily be answered in any other way. Normally this does not occur because we batch the requests. We have an absolute minimum of $20.00 for scanning the index file and master file, but at that point the cost curve rises very slowly. Some representative costs have been:

Records retrieved and sorted	Total Cost	Approximate cost per record
363	$20.02	5 cents
1,085	21.26	2 cents
1,812	22.66	1.5 cents
2,064	25.20	1 cent
3,642	30.54	1 cent
11,695	33.99	.3 cents
16,707	38.26	.2 cents
31,576	39.53	.13 cents
162,138	131.42	.08 cents

Print costs are in addition to these retrieval and sort costs. We are charged $.80 per thousand lines of print and $1.00 per thousand if special forms are used. Lines of print per record varies normally from 1 to 10. These costs are in general lower than those for the generalized data base systems we have tested. We do have worse cases occasionally when retrieval is on values not contained in the index file, but the same situation arises in Data Base Management systems.

I would like to refer back to some comments at the beginning of my talk. Should I be embarrassed by my dark ages system? Is being in the latest style more important than economy? Has there really been an evolutionary development from old fashioned sequential processing to Data Base Management? I think the answer to all of these questions is "no".

All of these techniques have existed in some form from the early days of computers. You choose what is appropriate to the problem. Example: In the early 1950's the John Plain Co. of Chicago[4] had a real-time inventory control system called "distribution". The data base was updated on-line, direct-access from 10 terminals. Batch reports were produced overnight. On-line inquiries could be performed. Example: The Atlas Supervisor[5] in use in 1960 had virtual memory, hardware paging, program roll-out and roll-back with relocation, I/0 data queues and request queues, an operating system that supervised many types of interchangeable I/0 devices. Example: At Griffiss Air Force Base a real-time inventory control system for spare parts was operational in the late 1950's on the Univac File Computer with direct access retrieval and updating. Standard batch programs were running simultaneously.

In conclusion, our experience can give a qualified answer to the question posed as the theme of the conference, "Innovative Systems: Solution or Illusion"? The answer is that it is an illusion to believe that most current data processing techniques have not existed for a fairly long time. They have all evolved and improved, including sequential processing. The solution, if you are interested in costs, is in careful analysis of requirements and selection of the most appropriate technique. The cost of maintaining and storing inverted files and pointer strings can exceed the costs for the Data Base itself. That may be acceptable if immediate access to on-line data is a requirement. The problem has been summed up by Alfonso F. Cardenas[6] in the Communications of the ACM, "The prevalent situation is that those schemes that have better access time performance have worse file update characteristics. As access time goes down, file update and storage requirements go up, and vice versa".

Beware of the innovator who is also a salesman.

Sequential File Maintenance Costs

This list shows the costs for processing a mix of add, change, and delete transactions against the master file of 200,000 records. "Total Cost" includes all sorting and print costs as well as the cost of the update job. The cost of the update program is also shown separately. The master file contains about 70,000,000 bytes of data, which in fully expanded records would amount to 223,000,000 bytes .

NR Transactions	Total Cost	Update Cost
1,909	69.32	39.10
2,562	88.02	54.62
3,072	89.22	50.83
4,618	91.18	42.97
5,515	110.46	54.03
6,621	124.60	59.37
7,540	105.20	46.20
8,740	122.73	49.92
9,083	123.46	52.22
12,303	140.72	55.52
14,083	147.71	75.47
16,240	167.01	55.83
20,602	174.33	52.14
22,345	235.58	63.58
28,459	245.20	60.69
61,141	446.34	113.23
83,896	554.75	138.27
106,512	693.79	169.52

In addition, the update of the index file with transactions generated by the main update program costs an average of $12.00 with a range from $10.00 to $30.00. The index file always contains the same number of records as the master file and consists of 27,000,000 bytes.

1. C. W. Bachman — "The Programmer as Navigator"
Communications of the ACM, November 1973, Vol 16 Nr 11.

2. Jean E. Sammet — "Programming Languages: History and Fundamentals"
P 11, Prentice-Hall 1969

3. Lewis Carroll — "Through the Looking Glass"

4. Lawrence P. Lessing — "Automatic Control"
A Scientific American Book, Simon & Schuster, 1955
P 64-66

5. T. Kolburn
R. B. Payne &
D. J. Howarth — "The Atlas Supervisor"
Proceedings of the 1961 Eastern Joint Computer Conference, P 279.

6. Alfonso F. Cardenas — "Evaluation and Selection of File Organizing - a Model and a System"
Communications of the ACM, September 1973, Vol 16 Nr 9.

PLANNING FOR A DATABASE

John K. Lyon
Manager
Data Base Applications
Honeywell Corporation

WHAT IS A DATABASE?

A database is the collection of all of the facts which are known about an enterprise. The facts exist because the enterprise which they describe exists, not because of a conscious effort to collect data. It exists in the machine readable records, pieces of paper in a filing cabinet, and as the redundantly stored elements of master files. The collection process occurs as a consequence of its normal operations. The organization of the data will determine how useful it will be in supporting the enterprise.

The enterprise is a formally organized entity which is created to fulfill a mission; that is, it has a purpose. The term "enterprise" is used rather than one which could imply a commercial organization. It should be obvious that a database is not something which applies exclusively to commercial organizations. An enterprise reacts in response to its own assessment of its needs. This assessment is made within the context of how well it is fulfilling its mission. The way in which it organizes itself, the assignment of responsibility, and the allocation of resources are done solely in response to this self-appraisal. Its response may not be the best one, but it is an internal action. The manner in which the enterprise views its data is a reflection of the way it organizes. If it treats its data as a resource, it will eventually arrive at the conclusion that the data belongs to the enterprise as a whole and not to the individual components. When this occurs and the enterprise begins to examine the nature of its resources, it will become aware that data should be managed. This is the beginning of the Database.

The database belongs to the enterprise. The users should think of it as integrated, even though it may be maintained on separate master files.

Database awareness is reflected in the discipline of standard edit criteria, reconciliation among files, and the ability to cross-reference between files. The conversion from an environment of independent master files to the database requires a management commitment to relinquish individual prerogatives which accompany independent and functionally oriented master files.

The database is the community resource which is the foundation for generation of information. It is data, not information. Information is the result of applying procedures to data. Data are facts which describe the enterprise and its environment. These facts are interrelated just as the parts of the enterprise. If the data are organized to reflect the information processes, as they usually are in master files, integration of data becomes difficult and expensive. Data should be organized to reflect the enterprise which it represents. Most of the change which can be observed within the information processing function is attributed to changing information requirements, not changes in the enterprise. Delays occur whenever someone wants a report format changed, requests information based upon the contents of more than one file, or would like a new sequence. Because a database considers the data as a community resource, its organization reflects the needs of all users and is not tuned to any single use or user.

Because the acceptance of the database concept involves change, its implementation must be planned in advance. The implementation process will disturb users who will be called upon to relinquish their private master files and submit to a shared environment. Integration requires that the contents of several master files which have been separately maintained be combined into a single file. This process will directly affect the users simply because it will force reconciliation of files.

The largest single expense in achieving a database environment will be the cost of reconciliation. An associated problem will be people. The introduction of a database will affect how people perform their jobs; it disrupts the status quo and forces users to rely upon someone else to gather and maintain data. It forces standards. The introduction of the database will require adjustments in the internal organization of the enterprise. The Database Administrator will be the first visible indication that something has changed. Subsequent changes can be expected as the responsibilities for data shift from the users to the DBA. Throughout the change in the way the enterprise operates there should be positive effects reflecting the ability to access data which has originated in different functional areas.

HOW DO I KNOW THAT I NEED ONE?

Unless the enterprise is new, it is likely that it has been able to survive without a database. In most cases, it is likely that it can survive a little longer by continuing that way. However, there are some indicators that an enterprise needs a database. They are not absolute but relative. The first indicator is the current systems: Are they adequate? Do they do what I feel they should be able to, or are they limited? A second sign occurs when an external agency imposes a new requirement as a condition for continuing the conduct of the enterprise's affairs. A database will be indicated if this new requirement forces the use of data that crosses the functional boundaries of the enterprise and is not available on a single master file. A third sign is the acceptance of the need for on-line processing. While on-line processing is not a precondition for the database concept, on-line processing usually crosses functional boundaries. This is equivalent to processing multiple master files. It is usually impractical to achieve acceptable performance using sequential master files. Because there is a requirement to process on-line, the existence of random access storage will encourage the implementation of a database. Finally, there is the existence of competition. There may not be a real business need for a bank to embrace on-line banking, but in order to retain a progressive image a bank may adopt the concepts. Because a competitor can provide "single account" service, I must also provide the external image of doing at least the same.

Current System Problems

The current information systems provide best indicators that an enterprise needs a database. Recognition of these problems is usually delayed until one of the other signs occurs or until there is new management needing information which cannot be provided using the current files and procedures. The existence of integrity problems is a direct consequence of the maintenance of multiple master files which contain the same data. Because the data is redundantly maintained by more than a single user and with more than one program, there will be differences between the files. But the files supposedly describe the same things. The enterprise is an integrated entity and its functions deal with the same things. They will report on them based upon the contents of their own files. But, the enterprise does not rely on one report. It relies upon the combinations of reports. Data integrity problems become visible when the data from more than a single file is compared. There are explanations as to how the differences could occur, but the presence of the differences requires resolution. Data discrepancies cause indecision and delay. They create doubt as to the validity of all information and can lead to the creation of even more private files.

If data is to be useful for producing information, it must be accessible. How many times have we heard the excuse that a report cannot be written for six months because the data is in the wrong sequence, the wrong format, or on the wrong file? The Database does not automatically resolve all of these problems, but it does make it more likely that customers on one file have the same customer-id on another file. The awareness that data is not private and that it is needed by someone else leads to the cooperative adoption of standards. The fact that users are aware that someone else will use their data leads to better documentation, not only of edit standards but the conventions which are reflected in their files (e.g., account 99999 is used to temporarily post suspense balances, customer number 1423 is used for internal transfers, etc.).

Responsiveness is a problem caused by private master files. How fast can the organization respond to the need for new information? Most of the problems of responsiveness are based in the inability to access data because it exists on separate master files and is structured to reflect the application and not the enterprise. The fact that the master files have their own separate update cycles leads to a narrow window in which the files are in phase.

Master files have traditionally been structured in terms of the application and maintained in a sequence tailored to the application. Sequential access methods lead to periodic updating of the files because the cost of processing the entire master file is large enough to discourage selective update. Even if the master file contains data which should be updated on a daily basis, the cost of update may lead to the selection of a weekly or biweekly update.

Sequential access costs are a function of the size of the master file and not the number of transactions to be processed. The database environment is usually based on random access techniques which encourage selective update and retrieval. If the information required to control the shipment of material from the factory is based upon the contents of a master file which is updated only weekly, the shipment data can be no better. Timeliness of information should be determined by the requirements of the enterprise and not the update cycle of a master file.

Redundant data has a real cost which is rarely measured in data processing. The cost of redundant data is not the cost of the extra storage media. It is not the cost which comes from a lack of data integrity. The real cost of redundant data is the cost of the resources used to update the multiple copies of the same data. These resources are processors, programmers, documentation, data preparation, data validation, and the protection of the data within their separate master files. This real cost is seldom identified because it is dispersed among the several application systems which use that data. In one case, a seven-part input document is burst and sent to seven different application areas. Each enters the data and each uses that data to update its master files. In addition, each area uses different edit criteria and different update cycles. Unfortunately, the form represents a requisition for materials. It should be obvious that this enterprise is paying too much, with the result that it has severe data integrity problems.

If an enterprise is faced with information system inadequacies, and if these problems are characterized by inconsistency, incompleteness, lack of timeliness or lack of responsiveness, the current methods of generating information should be examined to identify where consolidation could occur. Integration of data is the start of the database awareness.

External Influences

While the information systems currently in use should provide the evidence that a Database is needed, the stimulus to change frequently comes from outside the enterprise. One of the more common sources is the action of a customer, a regulatory agency, or legislative body. During the late 1960's, the Department of Defense instituted new methods for measuring progress on government contracts. Until this time, measurement was either on a time basis (Is it on schedule?) or a money basis (Is it within budget?). The new directive recognized that progress should be measured on both bases simultaneously. The new control was known as the Cost/Schedule Project Control System (C/S PCS). The effect of the directive was to integrate data that traditionally had been maintained by two separate functional organizations within the contractor's organization. Cost data traditionally had been maintained by the accounting organization, while progress in terms of schedule was a function of engineering or manufacturing. In order to comply with the requirements, many companies developed databases which would satisfy the new regulation. The external action broke down internal resistance to change.

All external influence does not come in this manner. The merger of two separate enterprises represents an action taken by the enterprise resulting in data integration. The data represents two separate entities with separate functional distributions of responsibility. In such cases, it is necessary to adjust the functional responsibility for data and the data itself. Because of the suddenness of the requirement, it provides the stimulus to integrate data. There are other forms of changes in the nature of the mission of the enterprise, less traumatic but also indicating the desirability to integrate data across functional boundaries. Typically, these will occur as its environment changes. For example, the presence of tight money may lead an enterprise to integrate in order to exercise more control over the enterprise; or the introduction of new weapons may indicate the desirability to integrate the command functions for a theater of operations rather than to maintain separate files to describe each component.

There are many cases in which the database awareness is stimulated by the acquisition of new Data Processing hardware. While the primary emphasis in selecting new hardware is not a desire to enter the world of databases, the existence of equipment and software which will support a database usually leads to its development. While the existence of a database capability was a major criterion used in the selection of hardware in the World Wide Military Command and Control System purchase several years ago, that requirement did not create any databases. It was the existence of the hardware and software at user sites that permitted the users to develop the databases.

In many cases, the stimulus to create a database has come as the result of the actions of a competitor. There are examples in the banking industry in which a bank with a large branch network selected and developed on-line systems as a business necessity. The database which resulted from the on-line systems was able to support other services for bank management (customer exposure, interrelationship of accounts, trust management, etc.). The presence of these services gave a competitive edge to the bank, which then stimulated its competition to follow suit. The reaction to competition is made without the corresponding direct business need. The same is characteristic of the airline industry, which has seen even the smallest of the regional carriers follow the lead of the major trunks in adopting on-line reservations systems. In many cases, it is questionable if the on-line systems would have been a sound investment if there had been no competition.

HOW DO I GET ONE?

The first step in implementing a database is to recognize what it is. The database is a resource. The resource is the data which is used by the enterprise to generate information. That information is the basis which the enterprise uses to make decisions. The database is the collection of facts about the enterprise; it describes the enterprise. But what is a description? It is a model. The database is a model of the enterprise and its environment. It describes the things within the enterprise which are important and the relationships among them.

If the database is accepted as a resource, and if it is to be useful to the enterprise, it should be organized and managed in order that it can be exploited to benefit the enterprise. In order to exploit it, the enterprise must organize itself to manage the resource. This organization of people and facilities into a function which has the responsibility for managing the resource is known as the Database Administrator (DBA). Just as the enterprise will formally organize to manage other resources, it must organize to manage data.

The Model

In developing the structure and organization of the Database, it is important to determine those things in the enterprise which are important. The modeling process is an examination of the enterprise and its environment. The purpose of the examination is the development of a logical structure of the database. In arriving at the structure, it is important to state and identify those entities which are the essence of the enterprise. For example, in the computer business there are two kinds of things which are central: customers and products. In the intelligence community, units and locations are central.

If you ask yourself the question "What is it all about?" you are trying to identify the central entities. If the definition of the enterprise is unnatural, if it is difficult to explain, it is likely that the truly important entities have not been identified.

The second aspect, related to the first, is a determination of what is needed to control the enterprise. Several years ago, we examined an Air Force application which was used to describe the data processing equipment and application systems in use within the Air Force. The existing structure was based on sequential files and was maintained with a primary sort key derived from central processor identifiers. During the discussions with the users and the data processing staff, it was implied that the system was controlling the use of data processing equipment within the Air Force. The structure of a database which would have replaced the sequential master files was unnatural and overly complex. After the second day, it became obvious that we just weren't getting anywhere. Finally, the questions "What is it all about?" and "What does the system do?" were asked. The answer was not equipment or labor or run times for application systems but where was the Air Force spending its data processing budget. The system was about dollars, not equipment. The use of central processor numbers as the key was misleading. The important entities, those central to the enterprise and how well it can fulfill its mission, will usually be the same ones which must be controlled.

Having identified the central elements, the remainder of the logical structure will usually fall into place with very little effort. This does not mean that the model is correct or that it will be implemented in this way; it just means that it is probably very close to the real world if it is built on the central entities and if it is natural. Naturalness is a quality of a good design, not just of database design. Naturalness implies simplicity. The best test of naturalness in a design is the ease with which it can be discussed and explained. If it is found that there are too many occasions on which the explanation becomes involved, cannot focus on one portion of the design at a time but hops back and forth and is too frequently qualified in response to the exceptions, the design is probably incorrect. A good design does not possess a long list of exceptions. A good design can accommodate the exceptions; it is built to handle exceptions, which have a habit of exceeding the normal conditions.

The existing systems should be used to test the model to see if it is sufficient to accommodate their data requirements. The existing systems are used as the basis for determining "what" is being done as a guide as to what is required to control the enterprise. The existing systems should not be the guide as to how it should be done. The systems are useful in determining where the current master files overlap and, consequently, where data reconciliation problems can be expected to occur. If the existing systems are adequately documented (functional requirement through program documentation), it should be possible to determine who is responsible for the accuracy and timeliness of the data.

The Organization

The current organization and the arrangement of the functional responsibilities within the enterprise will be affected by the integration of the data. Until integration has occurred, the functional components will operate with private files. This privacy permits designers and programmers to tune these files to the demands of the application systems and frequently to specific programs. This tuning is achieved through the arrangement of fields, the precomputation of field values, rollup of details, a specific sort sequence, and the content of records. The result of these techniques is frequently a system which runs more slowly than if it had not been tuned. The resulting inefficiency is due to sorting multiple master files, generation of redundant transactions, maintenance of bridge files, the proliferation of "master" files, and the attempt to respond to new requests for information. This suboptimization is often the result of a programmer's or designer's desire to make his job easier. Suboptimization is frequently substituted for good design.

Integration of data into a single file requires that the users relinquish their prerogatives over the master files which were theirs. It requires that they adhere to standards and schedules which benefit the enterprise. The loss of the ownership of a master file is frequently interpreted as a loss of importance. Integration is not basically a technical problem, it is a political problem. It crosses functional boundaries. It must fight NIH (not invented here), and "my problem is unique."

Integration of data can only occur if the enterprise is committed to the creation of a database and if that commitment can be sustained. It is rare for an organization to maintain its commitment if the original decision is not based on a response to a real need of the enterprise. Integration can be uncomfortable, it disturbs and disrupts. In order to achieve integration, it will be advisable to identify the DBA function, to staff it, and to give it the necessary responsibilities and budgets.

HOW DO I GET STARTED?

The first step in realizing a database is to develop an organization which can effectively interact with the users to develop a time schedule for the implementation. Because the database implementation is phased, this plan must be developed carefully to maximize the benefits and, at the same time, create minimum disruption to the operations of the enterprise.

There are two approaches to the development of a database implementation plan: a task force and a DBA. The solution which seems to work best is a combination in which the DBA works with a task force from the functional organizations to determine what the database should contain, where the data will be obtained, the impact on new system plans, and the schedule for implementation. A side benefit of this approach is the creation of a working relationship between the users and the DBA and the use of the task force as the focus for evaluating the database design.

Since the DBA will have the eventual responsibility for the actual implementation, it should serve as the leader of the effort to develop the plan. However, the DBA is the working leader. The development of the integration plan and schedule should be led by a member of the task force and should come from one of the user organizations.

The start-up phase will require that considerable thought and effort be applied to what the database will contain. This effort includes the development of the model, the logical structure of the enterprise. The involvement of the users will assure that resulting design represents the needs of the enterprise and is not limited to a single application area or the DBA's conception of the enterprise. The task force, as a representative sample of the functions, will serve as the first basis for evaluation of the design. The design will be expressed through a description of what the database will contain and what it represents. This description is used to elicit comment from the users, each with their own perspective of the enterprise and what is important. The description is not intended to be reducible to source code. It is meant to convey the intention and use of each record type, field, and relationship within the database and to indicate the implementation schedule for the different logical portions of the database.

HOW LONG WILL IT TAKE?

The length of time involved in the implementation of a database depends upon the enterprise. It will reflect the commitment that management has to the database and how well they have conveyed their commitment to the functional users. It will depend upon the state of the master files, how well they are maintained and documented, how well they reflect the enterprise, and the degree of consistency among the data. The design phase will reflect the complexity of the enterprise and its environment and, to a greater degree, how well the users know the enterprise.

It is not unusual to discover that the functional users and data processing staff do not really understand or agree on what the enterprise does, why it does it, or how it gets done. The design phase will usually be noisy and the design team must be careful that it is not used to elevate the importance of one function at the expense of another for artificial reasons. The development of the logical definition of the database (the model) will usually require about two months. The validation of the model, the repeated adjustment of its details, the rewriting and review of the description will require an additional three to six months. The biggest factor for the size of the time spread will be the cooperation of the users and the agreement on what is important to the enterprise. The development and acceptance of the implementation plan will normally require an additional two months. The actual realization of an implementation and the existence of a usable database will require at least six months and up to two years. If the process takes longer than two years, there is a very real risk that nobody will remember why it was done in the first place. The selection of the 2- to 2½-year timeframe is also related to the length of time that management remains stable.

The implementation phase will require the creation of a prototype requiring that users are prepared to devote resources to the reconciliation of data during this timeframe. The largest effort is that required to reconcile data during the integration of multiple master files. This is larger than usually assumed and has an impact on the enterprise. It requires resources beyond that under the control of the data processing organization and beyond that of the functional users who are directly involved with the database. It should never be minimized. Reconciliation is the largest single effort involved in the development of a database. The existence of the data dictionary to indicate the meaning of data, its sources, who is responsible, and where it is used will make the reconciliation effort easier, if only by reducing the effort in determining whom to talk to.

WHAT ARE THE PITFALLS?

The database does not come "for free." It will take hard work, and there are traps and pitfalls along the way. The first problem which must be confronted is NIH. The introduction of a resource which crosses functional boundaries is a people problem. The database will never succeed unless there is cooperation among the users and unless they use the database. There is always a lack of understanding as to why the database is needed and what it all means. The database contains data and relationships which are beyond the personal experience and knowledge of any one user. As a result, parts of it will represent a mystery.

The justification for a database is difficult to express in terms of tangible benefits. The savings in the new environment appear to be incremental to the users. The fact that data need be collected and maintained only once is tangible, but the savings to any single application is incremental. The determination of the magnitude of the savings is made difficult by the absence of a valid breakdown of the existing costs. The collection and maintenance costs for data are normally buried within the application system costs.

The effort required for reconciliation is extensive. It disrupts and disturbs the activities of functional personnel in order to arrive at a reconciliation. The rudeness of the need to reconcile does not make friends. It is a source of embarrassment to at least one party, if not more. The need for reconciliation raises valid questions as to why is it necessary, why is the data inconsistent, and what was the effect of decisions which were made based on this inconsistent data. In some cases, reconciliation may impact the reporting of the performance of the operations of the enterprise. It has happened that the reconciliation has caused a reduction in the value of inventory causing a restatement of earnings.

The existence of the database makes it possible to quickly alter the manner in which an enterprise can measure its performance. The existence of a set of subgoals and the knowledge of how they are measured has been known to bias the operation of organizations in order to achieve "good marks," when such actions may not be in the best interest of the enterprise. The database permits more flexibility and greater breadth in measuring performance. This is good but may meet with resistance. The database permits and requires that responsibility for data and its accuracy be defined. In defining this responsibility for data there will usually be a corresponding increase in the clarity in which other responsibilities are expressed.

The development of a database represents change. It is disruptive, it breaks down the security of functional boundaries. It requires effort and cooperation from the functional users. It will meet resistance, which is based on emotion rather than reason, which will make it more difficult to overcome. It and the motives of the people who were instrumental in its development will be distrusted. These factors impose a real need for a well developed plan and for good communication among the users. It requires that management be kept informed and that they not be given a "snow job." If the database is to exist as an integrated resource of the enterprise, it must have the continuing support of management and of the users.

WHAT ARE THE BENEFITS?

The primary benefits of the database are intangible. It is difficult to determine the value of a decision which is made with complete data or is based upon consistent data. How do you determine the incremental value of a decision which can be made in minutes or hours rather than in days because the data was available? There is increasing evidence that it is no longer a question of how much I can save through automation and integration but how can I survive without it. As enterprises strive to tighten their control over the use of their resources, integration is the only solution.

There are, however, real benefits. There is a real cost reduction which can be obtained by maintaining data only once. There is a real benefit to complete data when it permits the identification of lost revenue because all of the data was available. It gives real savings when unnecessary expenditures are eliminated because the data is timely and available.

The existence of a database permits an enterprise to respond faster to new requirements and to increase the flexibility with which it can evaluate its options. The database permits management to deal with the enterprise as a whole and not a collection of semi-autonomous entities. It permits management to manage.

* * * * * * *

PROCEDURE FOR ESTABLISHING DATA STANDARDS AT CORNELL

Lynne K. Personius
Data Base Administrator
Cornell University

An obvious prerequisite to data element control and to the effective interface between application systems and data bases is a complete set of data standards. At Cornell University, where many independent data systems serve the needs of diverse user groups, this process is complex. The procedure, made up of seven steps, begins with a management review of information systems inorder to define the direction of data standardization, and ends with a user review and acceptance of each final standard.

Step One, Overview.

The objective of step one is for the Data Base Administrator and management to arrive at an agreement as to the direction of the data review and standardization process, and the general format for data collection. This involves an overview of all existing application systems where the systems are roughly grouped according to purpose of system and type of data stored. The groups are then evaluated to decide which needs the most immediate attention, and are ranked accordingly.

At Cornell we assigned top priority to the group of systems containing personal information. These included Payroll/Personnel, Student Records, Alumni Records and Admissions. This group was the first to be considered in the standardization process. Group Two became systems containing financial data. We agreed that whenever an immediate need arose for a system in other than the preassigned groups to be standardized, an immediate need being new systems development or a new system interface, that the order of the processing be flexible enough to allow the new work to receive attention.

A general format for the standardization process was arrived at during the overview period. It will be defined in subsequent steps.

Step Two, Data Review.

Before any standardization process can begin, an extensive review must be done to document what data is currently being used in what systems. We began with data systems processing personal information. The first part of the data review process involved collecting summary level information about each system. A set of standard forms, contained in Appendix A was designed for this purpose.

The first form is a System Description form. It contains a general summary of the function of the system, a list of programs used and a list of all files processed in the system.

Its intent is to provide summary level information about each system, such that a collection of System Description forms will serve as a general index to Cornell information processing. Program Definition forms are completed for each program in the system. They contain a short description of the function of each program, and an enumeration of all inputs to and outputs from it. A File Narrative form is filled out for each data file processed by the given system. The information contained here includes kind of data stored, use of data, and method of storage. A Program Log and a File Log constitute an index to the Program Definitions and File Narratives. Collectively, the information obtained in this review process provides summary level insight into how data is being used and what data needs are being met by each information system.

At the same time as the general information described above is being collected, a manual Data Dictionary is built for that system. The format of the manual dictionary is included in Appendix B. The Dictionary provides specific information about each element used in the system. Such things as data field format, element name and definition are stored here. For data elements that are themselves codes, a translate table is included containing each data field value and its definition.

Without some form of Data Dictionary the standardization process would be impossible. The Dictionary defines the kind of information about each element that is needed inorder to develop a data standard. It provides a look at what elements are used, how they are stored specifically, and what code values they have. This information is compared, system by system, to determine variance between data definitions, values, and uses. The next step in the standardization process is dependent on having the dictionary completed for several systems using related kinds of information. At Cornell, we waited until all of Group One systems, those defined as containing personal data, had been reviewed and had their elements included in the Data Dictionary before developing the first standards. The first standards prepared were therefore those dealing with personal data.

Step Three, Establish Standardization Need.

Within Group One systems, more data elements exist than can be standardized at one time. The Data Base Administrator has responsibility for determining relative need for standardization and recommending priorities for the development of standards to management. Using the existing Data Dictionary the Data Base Administrator collects statistics on multiple uses of data elements among systems, and of discrepancies of data definition, format or code usage between the various information systems.

Arbitrarily, a determination is made of the most used, least standard data elements. The Data Base Administrator then prepares the necessary documentation to support a set of recommended priorities, and Step Four begins.

Step Four, Management Approval of Priorities.

Management must approve the priorities recommended by the Data Base Administrator. Inorder to do this effectively, the DBA must provide the information necessary to make a realistic judgement of relative need for data standards. The process that evolved at Cornell after many trial approaches is for the Data Base Administrator to provide a formal proposal to management. This proposal recommending the establishment of priorities would be divided by data element. For each element the following information is provided:

Number of systems using this data,
Name of each system,
Characteristics of the element in each,
Code tables used,
Recommended priority,
Special circumstances (example, needed in new system interface).

This information allows management to evaluate the priorities knowledgeably. The result of this step is a firm priority list for standardization.

Step Five, Draft Standard Prepared.

The Data Base Administrator prepares a draft standard referencing the previously built data dictionary. The draft is an attempt to resolve conflicts in data element usage between existing systems as simply as possible while at the same time arriving at the most workable standard available for existing and new systems. The format of Cornell Data Standards is demonstrated by the sample provided in Appendix C.

Some general guidelines were followed. When available and workable within the existing framework of information systems, Federal data standards were adopted. For example, the FIPS (Federal Information Processing Standard) codes were agreed to for geographic data. Not in all cases could Federal standards be used, however. An attempt to implement the occupation codes used by the U.S. Bureaus of the Census revealed that this code was too detailed and cumbersome for some Cornell applications. In some cases the data usage subscribed to by the majority of information systems was used as the draft standard.

In other cases only a totally new element definition could meet the various needs of differing systems. Where all systems used a different code, for example, and any could be adopted as the standard, the element definition used by the newest application became the draft standard. This was done under the assumption that older systems will be redesigned and reworked before newer ones, and that compliance to standards could be done at that time.

Step Six, Technical Approval

Technical approval of the draft standard is provided by two groups:

Senior Maintenance Applications Programmers,
Systems Analysts.

In the Cornell structure, Senior Programmers are responsible for Maintenance to existing application systems. They are therefore in an ideal position to assess the technical feasibility of a given data standard as it applies to existing systems. Systems Analysts evaluate the standard according to its workability in the design of new systems. Both groups are given adequate time to review and respond to every draft standard.

The Data Base Administrator reviews comments from the technical staff and attempts to resolve any conflicts. In some cases, the technical review results in standards being redrafted and recirculated because of something unforseen by the DBA. In other cases, the comments coming in are generally favorable. At that point the Data Base Administrator polishes the draft a bit more, and it is distributed to Users.

Step Seven, User Review and Acceptance.

So that Users can evaluate the effect any given data standard will have on their particular area, they are given certain relevant information at the time they evaluate the standard. Specifically each data standard is distributed accompanied by a form containing the following:

1. A list of which systems are already in compliance, and which are not.
2. An indication of whether conversion to this standard is required, suggested, or anticipated.

When special circumstances surround a given standard, more information may be appropriate either for all Users or for a particular group of affected Users. An example of such a special circumstance might be the case where a new system is being designed, and part of the new design is an interface between two existing systems.

Suppose the data to be shared in the course of this interface is not consistent between systems. The Users responsible for the particular systems should be made aware of this kind of problem when they evaluate the draft data standard.

Users are given a fixed amount of time to respond to any draft standard and to express objections, if they have any. If the given amount of time passes and no objections have been raised, the data standard is assumed satisfactory by the Data Base Administrator, and is finalized and implemented.

Objections to any draft standard are negotiated on a one-for-one basis with the Data Base Administrator. Any User who finds a draft standard not workable in his/her particular enviroment should notify the Data Base Administrator as soon as possible. When appropriate, draft standards can be revised and recirculated to reflect changes requested by Users.

Better than a majority of Users must accept a given draft data standard before it can be implemented. Once this has happened, the data standard is put into final form and is issued bearing the signatures of the Director of MSA and the Data Base Administrator. Each User receives a copy of the final document.

The process described above provides each User with the following:

1. A copy of each proposed data standard,
2. Relevant information needed to evaluate the standard,
3. Adequate time to evaluate the standard,
4. The opportunity to object and negotiate changes,
5. Notification of each final, accepted and implemented data standard.

Each User has the exposure to and participation in the data standardization process which is such an important part of Data Base Administration.

The procedure described here provides a systematic approach to data standardization in a complex environment. At Cornell, it began by enumerating all the existing information systems in an attempt to determine a logical starting point. Having determined where to start, we began a data review process. This required an enormous amount of time to accomplish effectively. We accomplished for the group of systems containing personal information the complete data review process. This material adequately supported the remaining steps in the standardization process.

Our procedure is still new. To date most of the time spent has been in the data review stage. We have decreasing numbers of standards at each stage in the process. We have taken approximately ten data standards all the way through the User review, and have numerous others in development. The procedure seems to work effectively in an environment where data needs are complex, and where multiple systems support independent and diverse User groups.

APPENDICES

Requirements for a "List of Cornell Systems".

A standard form must be completed for each system at Cornell. The following information is required:

1. System Name
2. System Code
3. User Department(s) - must be divided into those supplying input and those receiving input or output.
4. Generalized narrative covering the following:
   a. Purpose of system
   b. Description of operation performed
   c. Comments about system
5. List of files used by the system - divided into Master, input, internal and output.
6. List of programs operating in the system - identify uniquely.

A "system flowchart" must be submitted with each form.

CORNELL UNIVERSITY
SYSTEM DESCRIPTION

System Name ____________________ System Code __________________

User Departments (Indicate by each whether it supplies or receives input or output)

Generalized Narrative

A 2

CORNELL UNIVERSITY
SYSTEM DESCRIPTION

Files used by system (indicate by each either master, input, internal or output).

Programs Operating in System (identify uniquely)

A 3

Requirements for a "File Narrative of Cornell Files"

Each file or data base used as part of an administrative process at Cornell must be assigned a unique identifier. The format of the identifier will be:

'F' 9999T where:

'F' = file

SS = System code

9999 = unique sequence number

T = type, master (M) input (I), internal (N), output (O).

A master log of files and identifiers will be maintained by the Data Base Administration Function.

A form must be prepared for each file or data base at Cornell containing the following:

1. Identifier
2. File Name
3. Description of file, general
   a. Kinds of information stored
   b. Use of data
   c. Method of storage (device type, access method)

Included with each form must be a specific file layout, segment layouts for IMS files.

CORNELL UNIVERSITY

FILE NARRATIVE

Vol. I

Identifier ______________________ Name __________________________

General Description

Kind of Information Stored

Use of Data

Method of Storage (device type, access method, dbms)

Requirements for a "Program Dictionary of Cornell Programs"

A standard form must be completed for every program in the Cornell administrative computing environment. The form must contain the following information about each:

1. System affiliation, name and code
2. Program identifier
3. Program Name
4. Program language
5. Author and date-written
6. Names of sub-routines called (also, complete one form for each sub-routine)
7. Names of files processed.
8. Brief description - summarize purpose of program. Describe anything "special" about it.

A program identifier must be established for each program. It will be in the following format:

PSS9999L where:

P=Program

SS=System code

9999=unique numeric identifier

L=program language=

P=PL/1, C=COBOL, R=RPG, A=Assemble, F=Fortran, S=Snobol,

A log of these identifiers will be kept.

CORNELL UNIVERSITY

PROGRAM DEFINITION

Identification

Program Name	____________	System Name	____________
Program Identifier	____________	System Code	____________
Programming Language	____________	Author	____________
Date Written	____________		

Sub-routines

List called sub-routines

Files Processed

List all files used by program

Description

Page ______

Log Sheet

Master Files, Data Bases, and Identifiers

dentifier F'SS9999T	File Name	System Name

Page ______

Log Sheet

Program Identification

Identifier 'P'SS9999L	Program Name	System Name

CORNELL UNIVERSITY
DATA ELEMENT DICTIONARY

APPLICATION AREA

ELEMENT NAME

SYSTEM

SEGMENT NAME

DATA BASE

AREA NAME

IMS CROSS REFERENCE

CHARACTERISTICS

Min Max LENGTH

PROCEDURE NUMBER

DEFINITION

Aver. Max OCCURRENCE(S)

HOME SEGMENT

PARENT SEGMENT

PROGRAM PROCESS & FUNCTION

1

2

3

4

5

6

REPORT I. D.

1

2

3

4

5

6

OFFICE RESPONSIBLE for DATA ORIGIN

FIELD CODE

SECURITY RATING

CODE TABLE

SAMPLE

Management Systems and Analysis

Data Standard 2

Standard Name Information

1. Name of Standard. Standard Name Information

2. Category of Standard. MSA Data Standard.

3. Explanation. This standard provides definitions for the fields of an individual's full legal name.

4. Specifications. Full legal name is made up of the following fields:

    a. Last name, twenty-two alpha numeric characters.

    b. Suffix, three alpha numeric characters.

    c. First name, fifteen alpha numeric characters.

    d. Middle name, fifteen alpha numeric characters.

5. Qualifications. None.

6. Special Information. None.

USING THE ADABAS SYSTEM

David Benevy
Chase Manhattan Bank

USING THE ADABAS SYSTEM

1. INTRODUCTION

Chase is a large, full service commercial bank, offering a comprehensive set of banking services. Currently these services are run on a separate accounting systems, which together process approximately one million transactions a night. These systems, although capable of meeting basic operational requirements, fail to provide the type of access to their data that would result in the realization of the full potential uses of this data.

1.1 New System Requirements. To make more effective use of the Bank's data the current accounting systems would be replaced with a new system which would:

- Process the Bank's operational data
- Provide the logical capability of retrieval of customer information on multiple keys.
- Provide marketing support facilities: e.g., by providing the capability of selecting subsets of the data base.

1.2 System Constraints. In designing a system to meet the functional requirements identified above, the following constraints were imposed:

- The existing accounting systems must be easily convertible to run under the new system.
- Further system development and on going maintenance costs must be minimized.
- Mass storage utlization must be effective.

2. CUSTOMER INFORMATION RETRIEVAL SYSTEM (CIRS)

CIRS satisfies the new system requirement of providing on-line retrieval of customer information. CIRS will replace the current manual account locator system, which provides customer name to account number, and account number customer name translations. CIRS will be available for on-line use by branch personnel, as well as by computerized systems. In addition to providing the translation facilities of the current system, CIRS will also be able to reference any data item maintained in the customer's accounts. The information for any customer in CIRS will be available at any branch.

2.1 Branch Personnel Use. CIRS would be available for use by branch personnel. The system could be used in the following ways:

2.1.1 Routine Customer Query. To provide the customer with routine but infrequently asked information, such as an account number or the late fees due on a loan payment. Tellers would have access for this purpose.

2.1.2 The Customer Snapshot. CIRS could provide branch officers with a snapshot of a customers activity. This would be most effectively done with a CRT. This information would let the officer know his customers better and promote the sale of new services.

2.1.3 Overdraft Handling. The Customer Snapshot could also be used when the customer is not in the branch, for example to handle checks returned for insufficient funds.

2.2 Computerized Systems Use of CIRS. A good application for CIRS is One Statement Banking, where Savings, Loan and BankAmericard data appear on a checking statement. To request these summaries, the customer must supply the Bank with the account numbers. The checking account number and statement preparation date are recorded in the other systems, which supply the checking system with summary information for the checking statement.

With CIRS, automatic One Statement Banking will be possible. The checking system will be able to determine other customer accounts and extract the information itself. This will decrease customer involvement, Bank paper work, and the complexity of the related systems.

3. THE ENTITY AND RELATIONSHIP MODEL OF THE BANK

The operation of the Bank can be viewed as the maintenance of relations between itself and entities. The most common entity is a customer, and the most common relation is a customer's account. A given entity can have many relations with the Bank. For example, John Doe can have a Savings and Checking account, be an employee of the Bank and own our stock in the corporation. We identify John Doe as an entity, and associate with this entity the relations above.

In general an entity does business with the Bank and a relation is the type of business transacted. Individuals, unincorporated businesses, corporations and governments are examples of the entities with which the Bank deals.

3.1 The Universal Entity File (UEF). The core of the proposed system is the Universal Entity File. All entities with which the Bank does business are represented in this file. Each entity has a record containing its basic descriptive information; name, address, identifying numbers, etc. In addition, the entity's relations are represented as pointers to records in other files containing detailed information on the relation. Each entity is identified by a unique number, known as its Universal Entity Number (UEN).

3.2 The Relationship Files. Each type of relationship maintained by the Bank is assigned a file where data unique to this relationship is stored. For a Savings account this would include the customer(s) UEN(s), current balance, interest paid to date, date account opened, etc.

4. THE DESIGN OF THE UEF

In its basic form the UEF consists of fields describing the entity and pointers to records in the relationship files owned by this entity.

UEN	Name	Address	Various Descriptive Fields	Savings Acct #	Loan Acct #	Employee Id #

The record format above is straight forward and easy to use. This is so because all possible relation and descriptive fields are represented in a single record.

Since most of the fields in any record will be zero, a potential disadvantage of this approach is considerable wasted storage. The only way that this structure can be economically feasible is if the storage for empty fields can be suppressed. This capability then is a mandatory functional requirement of any DBMS to be used in implementing this design.

4.1 UEF Design Alternative. The sparseness of the proposed data structure is due to the maintenance of all possible relationships and descriptive fields in a single record. Other organizational strategies can be employed to reduce storage requirements, but usually only with an increase in the complexity of the system.

The maintenance of descriptive fields for all types of entities results in additional empty fields, since different entity types will require different descriptive fields. In hierarchical systems, additional branches of the tree can result in a reduction of storage space if the elements of a branch are chosen so that all will be empty together. Descriptive fields used only for corporations could be placed in a separate branch, so non-corporate entities would not have to allocate storage for them. A disadvantage of this approach is additional programming complexity, because the programmer must take into account the full data base structure.

An alternative solution to the descriptive field problem is to maintain separate files for different entity types. This approach increases programming complexity when objects have to be selected from both files.

The storage required to represent all possible relationships can be minimized through chaining techniques. By maintaining a chain of the entity's relationships storage is allocated only for active relationships. This approach entails greater program complexity for updating the chains as new relationships are entered into and old ones dissolved.

5. THE DESIGN OF THE RELATIONSHIP FILE

Records in relationship files have the following general format:

Account Number UEN Fields for the Relationship

The account number is the primary identification of the record. The UEN identifies the entity that owns the account. The remaining fields are the operational data of the relationship. The account numbers in the UEF correspond to the account numbers in the relationship files. All information describing the entity is maintained in the UEF record for the given UEN. This results in reduced storage for relationship files, and a reduction in data redundancy across the entire system.

6. CHOOSING A DBMS FOR THE SYSTEM

To facilitate the rapid conversion of existing accounting systems, it would be desirable if the information retrieval aspects of the new system, as opposed to simple transaction processing, could be superimposed over the existing systems with little of no knowledge that this was being done by the programmers responsible for these systems. Also, because of the quantities of data involved, it is mandatory that the DBMS provide a strategy, other than a full file scan, to resolve a query. Whatever strategy is used to optimize query times, additional data will have to be maintained for this purpose. In order to minimize new developmental costs, and permit an economical conversion of the existing systems, we require that the DBMS itself be responsible for the creation and updating of this data.

To minimize the developmental costs of new applications which use the querying capability of the system, we require that the determination and implementation of data base search strategies to solve a query also be done by the DBMS.

All the requirements above are met by the inverted list capability of the ADABAS system.

We have seen in the design of the UEF that the suppression of empty fields results in a simplification of the system design and substantial savings in disk space. ADABAS is the only available

DBMS offering such a facility. ADABAS also offers other strategies to reduce disk space requirements.

The ease with which ADABAS can process queries, combined with its data compaction capabilities indicate that it is functionally the best DBMS for this application.

7. THE ADABAS SYSTEM

ADABAS stores data in files as variable length records, comprised of variable length fields. There are no chains, pointers nor links within the data storage area. A separate storage area referred to as the ASSOCIATOR maintains inverted lists to data.

The three most important features facilitating the maintenance and retrieval of data are:

- File Inversion
- Record Compaction
- File Coupling

7.1 File Inversion. ADABAS can create an inverted list of any field in a file. These lists are used for random single key retrieval of records and for the resolution of complex queries involving several keys. The fields which have inverted lists are called DESCRIPTORS.

7.2 Record Compaction. ADABAS offers three different techniques to reduce the storage required to store a record.

7.2.1 Compression of Fields. Compression results in the removal of trailing blanks from alphabetic fields and leading zeros from numeric fields. This is useful for fields that are highly variable in length, such as customer's name and address or an account balance.

7.2.2 Suppression of Empty Fields. Suppression of empty fields results in the full suppression of the second and subsequent consecutive empty fields in a record.

The availability of this technique is prerequisite to the design of the UEF. In addition it offers substantial savings where we offer services within a service. For example, Cash Reserve checking accounts allow the customer to write checks larger than his balance. Since many customers do not have this feature, empty field suppression allows us to minimize storage for their accounts.

7.2.3 End of Record Field Suppression. When empty field suppression is used for fields at the end of a record, not even the single byte with the count of empty fields is stored. Consequently, the definition of a file may be modified to include new fields, (which are initially empty until the records are updated), without any reorganization of the data base.

7.3 File Coupling. File Coupling is a method for relating files on a common field. It will be explained by use of an example.

Consider a file of records licensed drivers and a file of records of car owners, each record with a social security number field.

Licensed Driver File

Name	Address	Zip Code	License #	SSN

Car Owners File

Name	Address	Zip Code	Car Make	Model	Color	Year	SSN

These two files can be coupled on their social security number field. When two files are coupled there is a list for each record in each file of the records in the other file which have the same value of the coupling field. These lists are used internally by ADABAS to resolve queries involving the two files. Such a query might ask to find licensed drivers living in Manhattan who owned Cadillacs built in 1970.

Many files can be involved in coupling relationships. In addition to the files described above, consider a boat owners file with a social security number field.

Boat Owners File

Name	Address	Zip Code	Boat Make	Model	Year	SSN

This file could be coupled to the licensed drivers file using the social security number field. We could then solve queries involving the three files such as finding licensed drivers owning 1973 or newer cars, who also own boats.

8. IMPLEMENTATION CONSIDERATIONS FOR THE UEF

The basic design of the UEF must be extended to provide support for unusual cases allowed by the current system. The current system imposes no restrictions on a customer having more than one account of the same type. Since all data for an account is stored in the single record for the account, a given customer can have several different mailing addresses. Sometimes different addresses can be due to minor spelling differences, but at other times the difference is desired by the customer.

The multiple field feature of ADABAS can be used to solve these problems. A multiple field can have a variable number of occurrences. The address field and those relationship pointers that can occur more than once are defined as multiple fields. To indicate which occurrence of the address field is to be used for a particular relation we define an address locator multiple field for each relation. The address locator fields are paired one to one with their corresponding relationship fields. The value of the address locator field indicates which of the address lines is to be used for this occurrence of the relationship. To conserve space we agree that a zero value will indicate that the first line should be used.

Name	Address	Descriptive Fields	Savings	Checking	....	Savings Address Locator	Checking Address Locator

For the great majority of entities, a single address will be sufficient, and all of the address locator multiple fields will be empty. Since they are at the end of the record they will not occupy any storage.

9. IMPLEMENTATION CONSIDERATIONS FOR THE RELATIONSHIP FILES

The most important consideration in the design of the relationship files is the ability to process a high transaction volume. To do this, the relationship files are loaded in account number sequence, and transactions are sorted in this sequence. ADABAS uses blocks of 3140 bytes, and we expect from 15 to 45 accounts per block. When there is more than one account with a transaction in each physical block, the file update run will require the I/O transfer of the entire data storage area and the decompression of those accounts with transactions. In this case, the elapsed I/O time depends on the quantity of data in the file, not on the number of accounts with transactions. The placement of descriptive customer information in the UEF, combined with data compression in the relationship file decreases the size of relationship records and consequently results in improved processing times. Information in the UEF is rarely accessed during transaction processing, so there are no serious consequences of it not being in the relationship files.

To enable querying, some fields in the relationship files are made descriptors. The choice of these fields will affect system performance and query capability. The best descriptors are those which enhance the query capability and change infrequently. A Cash Reserve Line is such a field. The worst descriptors are those which change with each transaction. An account balance is such a field. A method of minimizing associator updates while still maintaining query capabilities to the field is to create a range field. This is a descriptor field that indicates a range that the balance lies in. Queries based on dollar amounts are usually of the form "all accounts with balances over $500" rather than "all accounts with a balance of $772.33.". The actual amount specified is also probably not very critical. A further reduction of associator updates is possible if the ranges overlap. Overlapping ranges address the problem of accounts which fluctuate between two ranges. With overlapping ranges, $250-$1000 might be one range and $900 - $2500 would be the next. A balance starting in the first range, that later went to the second, would not return to the first until it went below $900.

10. COUPLING THE SYSTEM

We have shown how ADABAS could process the Bank's transaction workload, and the Customer Information Retrieval System. These activities access a few records at a time. Transaction processing requires a single record in a relationship file. The Customer Information Retrieval System at most requires a record in the UEF, and the associated records in the relationship files.

To be an effective marketing tool, the system must be able to select groups of customers based on fields in the UEF and relationship files. ADABAS can perform such selections automatically if the UEF is coupled to the relationship files on the basis of account number.

To illustrate the use of coupling consider cross selling automatic deductions from checking accounts to pay loan accounts. A good collection of customers for this service would be those with checking and loan accounts, without the automatic payment service, who have paid late fees on the loan account. With file coupling, this subset of the entire customer data base could be determined with minimal programming and processing time.

DATA DICTIONARY SYSTEMS

A. H. Cassidy
Assistant to the Director
Information Systems Department
University of Toronto

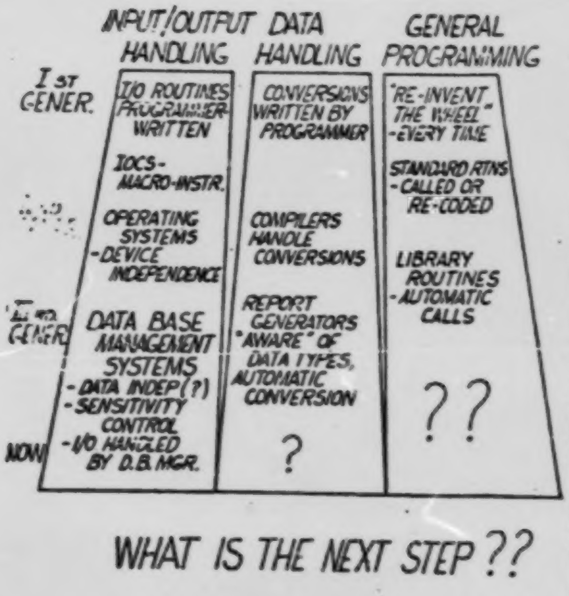
UNIVERSITY OF TORONTO

466

DATADICT

DATA DICTionary SYSTEM

TRENDS


INPUT/OUTPUT HANDLING
DATA HANDLING
GENERAL PROGRAMMING
1st GENER.
I/O ROUTINES PROGRAMMER-WRITTEN
IOCS-MACRO-INSTR.
OPERATING SYSTEMS -DEVICE INDEPENDENCE
DATA BASE MANAGEMENT SYSTEMS
-DATA INDEP (?)
-SENSITIVITY CONTROL
-I/O HANDLED BY D.B. MGR.
CONVERSIONS WRITTEN BY PROGRAMMER
COMPILERS HANDLE CONVERSIONS
REPORT GENERATORS "AWARE" OF DATA TYPES, AUTOMATIC CONVERSION
?
"RE-INVENT THE WHEEL" -EVERY TIME
STANDARD RTNS -CALLED OR RE-CODED
LIBRARY ROUTINES -AUTOMATIC CALLS
??
3rd GENER.
NOW

WHAT IS THE NEXT STEP??

Vol. I

The University of Toronto's Information Systems Department has developed a system to aid in the control and documentation of database-oriented systems. This system is called DATA DICTionary, and was developed in 1971 primarily for use in the student's information systems area. It has since been adopted for documentation purposes in other areas.

To evaluate the position of a data dictionary system in the scheme of things, one may examine some trends that have been in effect in information processing; (1) a trend towards data independence, that is, towards the indifference on the part of the applications programmer as to precisely how data is stored; (2) a similar trend towards format independence, i.e., different data types; and (3) a growing realization that there are commonalities in the area of general programming.

COMMON SOLUTIONS

- DB-ORIENTED SYSTEMS
  - APPEAR COMPLEX
    BUT:
  - ARE REALLY A SERIES OF STANDARD OPERATIONS:
    - EDIT
      - VERIFICATION OF DATA VIA TABLES, RANGES, CODES
      - BATCH BALANCING
      - TRANSACTION REFORMATTING
    - UPDATE
      - INSERT
      - DELETE
      - REPLACE
      - SPECIAL ROUTINES
    - RETRIEVAL
      - SELECTION STRATEGY
      - DATA EXTRACTION
      - REPORT GENERATION
- STANDARD METHODS
  - PERMIT SHARING OF PROGRAMMING LOAD
  - PROMOTE UNIFORMITY

Although each database-oriented system may appear to be both unique and complex, a certain analysis will reveal that the complexity is largely illusory, and that different systems do indeed have major components in common. Developing standard solutions to these common problems can aid significantly in the implementation of DB-oriented systems.

MANAGEMENT OF CHANGE

- A FACT OF LIFE
  - SYSTEMS ARE IN CONSTANT CHANGE
  - EXACT CHANGES NOT PREDICTABLE
    BUT:
  - DIRECTION AND DIMENSION OF CHANGE CAN BE PLANNED

NEEDED:
A "FLEXIBLE" SYSTEM
- PERMIT DATABASE EVOLUTION
  - STRUCTURAL CHANGES
  - EXPANSION
  - NEW SEGMENTS
- PERMIT FAST RESPONSE TO CHANGES IN ENVIRONMENT
  - TRANSACTION FORMATS
  - REPORT FORMATS
- DOCUMENT THE CHANGES

Change - like death and taxes - is among the few certainties in the life of a system. Although change cannot be avoided, its effects can be minimized by predicting the dimension and the direction of change, and by providing a flexible system to handle changes in the expected areas - database evolution and rapid changes in the environment. Most importantly, some method should be formed to document the changes automatically as they occur.

DATA DICTIONARY

- ORGANIZED FACTS
  - DATA BASE STRUCTURES & FORMATS
  - DATA ELEMENT CHARACTERISTICS
  - TRANSACTION FORMATS
  - REPORT FORMATS

AND:

- METHOD OF EXTRACTION
  - DOCUMENTATION
    DATA HANDBOOK/CODEBOOK
  - DBD/PSB CREATION
  - EDIT/UPDATE CONTROL
  - INFORMATION RETRIEVAL CONTROL
  - REPORT GENERATION
  - CONVERSION PGM GENERATION

The solution offered for all these needs is the DATA DICTIONARY. This is defined as a set of organized facts about the system - the structure and format of the data bases, data element characteristics, editing, etc., and formats of transactions - plus a method of extraction of usable intelligence about the system, such as human-readable documentation and machine-readable control information.

AIM

- TO DEFINE
  - SYSTEM ELEMENTS
  - RELATIONSHIPS
  - RULES & CONTROLS
- TO CONTROL
  - EDIT PROCESS
  - UPDATE PROCESS
  - INFORMATION RETRIEVAL
- TO COMMUNICATE
  - HARD COPY DOCUMENTATION
    - COMMUNITY OF USERS
    - PROGRAMMERS AND ANALYSTS
    - OPERATORS
  - CONTROL INFORMATION
    - IMS
    - APPLICATION PROGRAMMES

The basic aim, then, of the data dictionary system is threefold: (1) to DEFINE the elements of the system and their relationships, rules, and controls; (2) to actually CONTROL, via separate control datasets, certain common computer processes; and (3) to COMMUNICATE with the users of the system in human-readable or machine-readable form.

OBJECTIVES
– SYSTEM DESIGNER

- DESIGN AID
  - – DOCUMENT DESIGN DECISIONS
  - – SIMPLIFY DESIGN OPTIONS
- DOCUMENTATION AID
  - – REDUCE DOCUMENTATION LOAD
  - – PROVIDE CROSS-INDEXED MANUALS, ETC.
- REVISION AID
  - – DOCUMENTATION UPDATES
  - – REDUCE OR ELIMINATE CHANGE PROGRAMMING
  - – AID IN D.B. CONVERSION
- OPERATING AID
  - – STANDARD OPERATIONS

The prime user of the data dictionary system is the chief designer of the target system. The objectives of DATADICT from his point of view are to help him in the design stage by recording design decisions and simplifying options. More tha this, it should reduce the human task of documentation, and should assist considerably in the inevitable revisions. Finally, it should reduce the operator education required by promoting standard operations.

OBJECTIVES
– D.B. ADMINISTRATOR

- CONTROL MECHANISM
  - – D.B. DESIGNS
  - – DBD'S/PSB'S
  - – REVISIONS
- CENTRAL REPOSITORY
  - – SYSTEM:SYSTEM COMPATIBILITY
  - – CENTRAL DATA INVENTORY
- STANDARDIZATION AID

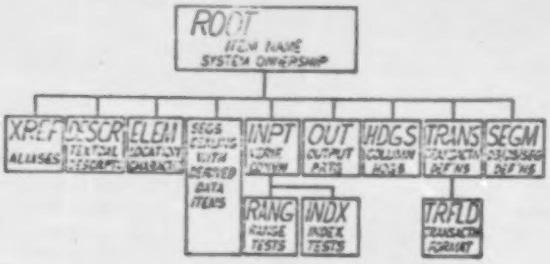
The second major user of a data dictionary system is the data base administration function. Such a system should provide a mechanism for the DBA's control of data base designs, the security scheme to permit access, and revisions thereto. It also would incorporate the DBA's most important tool, the central data inventory. Standardization and inter-system compatibility objectives are also served.

PRINCIPLES & CONCEPTS

- SYSTEM "OWNERSHIP"
  - BOUNDARIES OF SYSTEMS DEFINED
  - PHASE I - SEPARATE DICTIONARIES
  - PHASE II - COMMON DICTIONARY WITH SYSTEM BOUNDARIES
- SINGLE-ENTRY
  - SYSTEM ELEMENTS DEFINED ONCE AND REFERENCED WHERE NECESSARY
  - UPWARD REFERENCES
    - . AUTOMATICALLY INVERTED
- INDEXING
  - K.W.I.C. (KEY WORD IN CONTEXT) INDEXING, COMBINED WITH:
  - ITEM REFERENCE INDEXING

The DATADICT system operates on a few basic principles and concepts: (1) The concept of system "ownership"; that is, all elements belong to one system or another. In the strict sense, there are no "shared" elements. (This is implemented in the current version by separate dictionaries for each system; in the future, these will be integrated into a single dictionary with system boundaries incorporated). (2) The concept of "single-entry"; that is, each element is defined only once and then referenced where necessary and only "upward" references need be made - these are subsequently inverted by the system. For example, each data element in a segment refers to the containing segment name; the system then inverts these to give a list of data elements in the segment. (3) The indexing technique; a Key-word-in-context (KWIC) index is constructed on the first line of the textual description of each element, and this is merged with an item reference index, both for main reference and subsidiary references. This combined index, although large, provides a single reference source for information about the system.

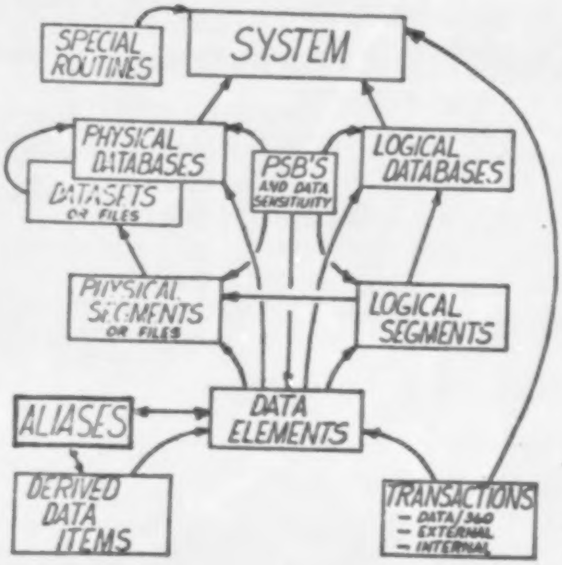
DATADICT D.B. STRUCTURE



- IMS DATABASE
- ALIAS SYSTEM FOR DATA ELEMENTS
  - BI-DIRECTIONAL POINTERS
  - AUTOMATIC MAINTENANCE
- KWIC-INDEX ON FIRST TEXT LINE
  - MERGED WITH ITEM INDEX

The DATADICT information is stored in an IMS database with the structure shown. Various system elements use different sets of segments, as will be shown later. The DESCR segment is of common use - this contains textual description of the element, and the first line should contain a brief title for the element, as it is used as the basis for a KWIC-index. Data elements also may contain XREF segments supporting a system of alias names.

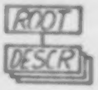
SYSTEM ELEMENTS



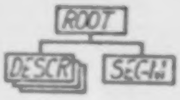
The array of element types in a typical system are shown in this diagram. The arrows indicate references (which are automatically inverted to form an owned list.) From the SYSTEM, the structure flows in two streams, for physical and logical DATABASES, DATASETS, and SEGMENTS, to rejoin at the DATA ELEMENT. Additional items are DERIVED DATA ITEMS (with ALIASES), TRANSACTIONS, PROGRAMMES, and SPECIAL ROUTINES. These are briefly described in the following charts.

SYSTEM ELEMENTS (I)
SYSTEM, DATABASES & SEGMENTS

- SYSTEM
  - TEXTUAL DESCRIPTION ONLY



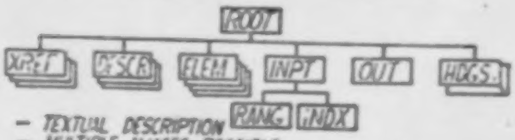
- DATABASES
  - TEXTUAL DESCRIPTION
  - 'HOME' SYSTEM, ACCESS METHOD, INDEX, ETC.



- DATASETS "
  - TEXTUAL DESCRIPTION
  - CONTAINING DATABASE, DDNAMES, RECORD & BLOCK SIZES, ETC.
- PHYSICAL SEGMENTS "
  - TEXTUAL DESCRIPTION
  - CONTAINING DATASET, PHYSICAL & LOGICAL RELATIONSHIPS, INDEX SEG, LENGTH, AND POINTER OPTIONS
- LOGICAL SEGMENTS "
  - TEXTUAL DESCRIPTION
  - CONTAINING DATABASE, PARENT, SOURCES, LENGTH

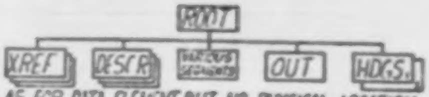
SYSTEM ELEMENTS (II)
DATA ITEMS

- DATA ELEMENTS (PHYSICALLY EXISTENT)



  - TEXTUAL DESCRIPTION
  - MULTIPLE ALIASES POSSIBLE
  - MULTIPLE LOCATIONS IN PHYS & LOG. SEGMENTS
  - INPUT VERIFICATION & CONVERSION REQMTS.
  - OUTPUT PRINTING CHARACTERISTICS
  - COLUMNAR HEADINGS FOR REPORTS

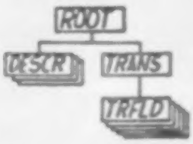
DERIVED DATA ITEMS



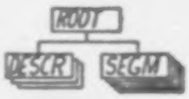
- AS FOR DATA ELEMENT, BUT NO PHYSICAL LOCATION
- DERIVED FROM DATA ELEMENTS
  - TABLE LOOKUP/FUNCTION REFERENCE
  - PRIORITY LIST
  - CONCATENATED NON-CONTIGUOUS DATA
  - TRUNCATED SUBFIELDS
  - LOGICALLY CONDITIONAL ITEMS
  - SEGMENT OCCURENCE COUNTS

SYSTEM ELEMENTS (III)
TRANSACTIONS PSB'S AND SPECIAL ROUTINES

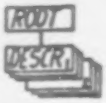
- TRANSACTIONS
  - DATA/360 (ADAPTABLE TO VIDEO/370)
  - EXTERNAL (INPUT TO GENERALIZED EDIT PGM)
  - INTERNAL (INPUT TO GENERALIZED UPDATE PGM)



- PSB (PROGRAMME SPECIFICATION BLOCK)
  - INFORMATION ABOUT:
    - LOGICAL TERMINALS
    - DATABASES USED
    - SEGMENT SENSITIVITY
      - PROCESSING OPTIONS
    - DATA SENSITIVITY



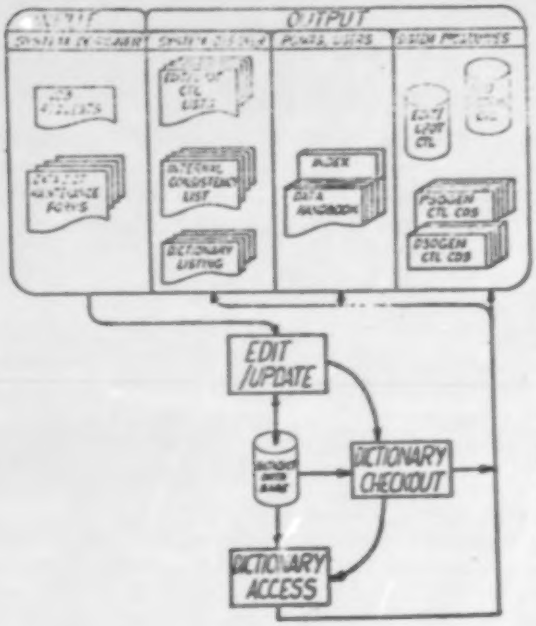
- SPECIAL ROUTINES
  - BASICALLY TEXT ONLY
  - CAN CONTAIN PL/I TEXT OF ROUTINE IN SPECIFIED RANGE OF LINE NOS.



These charts describe the structure of the various system elements in the dictionary ROOT and DESCR segments are common; XREF is used for data items, the SEGM segment is used in the description of databases, datasets, segments, and PSB's. TRANS and TRFLD segments are used for transaction definition.

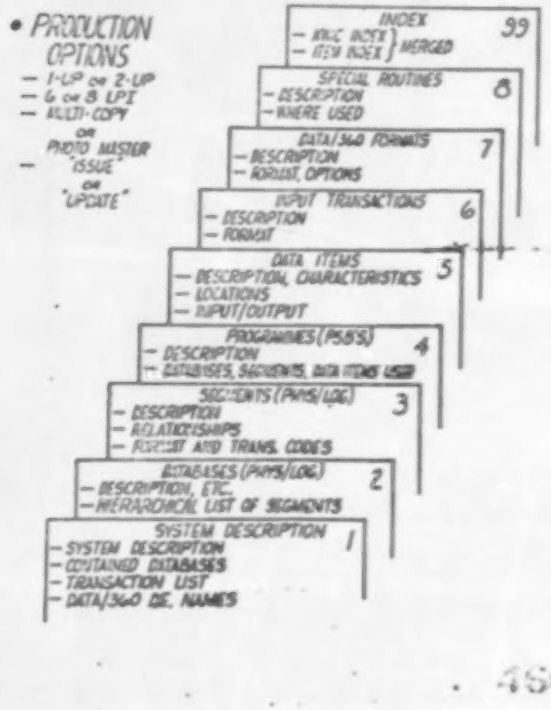
Vol. I

PROCEDURES



The procedures used by the system designer or other user in the DATADICT system are grouped into 3 categories: (1) Edit and Update of the dictionary; (2) Dictionary Checkout, that is, listing the dictionary and Internal Consistency Check, and (3) Dictionary Access for Handbook, DBD/PSB, Edit/Update Control, or Information Retrieval Control. The typical process is to build or maintain the dictionary via (1), perform some additional checks via (2), and then acquire the desired output via (3).

DATA HANDBOOK



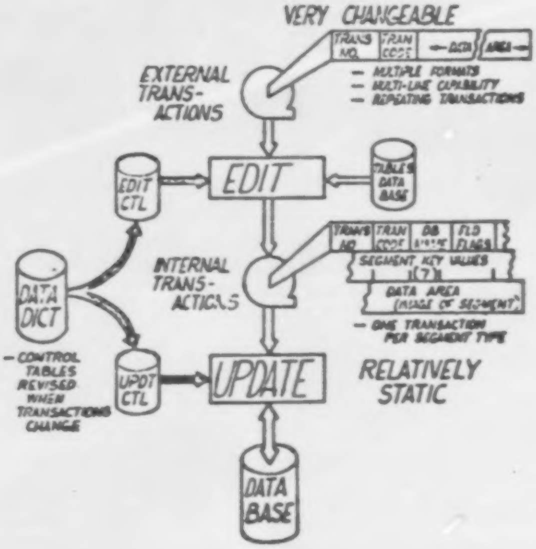
The Data Handbook is the major human-readable output of the DATADICT system. It is divided into nine sections, for eight categories system items plus the merged KWIC/Item index. The Handbook may be produced in "ISSUE" mode - entire handbook is produced - or in "UPDATE" mode, where only altered pages are produced.

DBDGEN/PSBGEN CREATION

- DBDGEN
  - PRODUCES CARDS READY FOR INPUT TO IMS UTILITY DBDGEN
  - OPTIONS: ALL "FIELD" CARDS OR ONLY KEY FIELDS
  - HIDAM INDEX DBD ALSO PRODUCED
- PSBGEN
  - USED FOR BATCH OR MSG PROCESSING PROGRAMME PSB'S
  - ADDITIONAL: PROVIDES DATA FIELD USAGE CROSS-REFERENCE

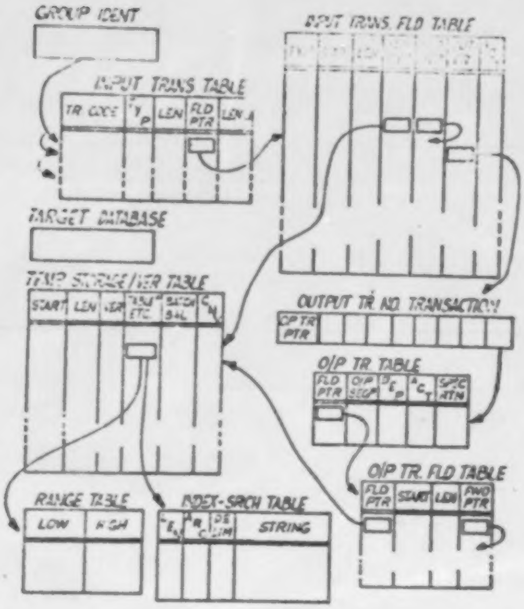
As a support to IMS, DATADICT also can produce input card deck for the IMS Data Base Description Generation (DBDGEN) or Programme Specification Block Generation (PSBGEN), with the options and features as shown on the chart.

EDIT/UPDATE CONTROL



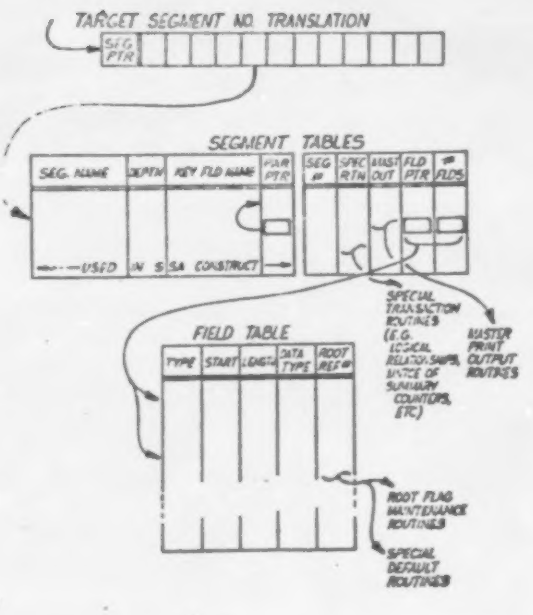
One of the unique aspects of the DATADICT system is its concept of a generalized, table-driven Edit/Update for the target system. The Edit may be seen as the interface between the volatile, changing environment of the external transaction formats, and the relatively stable domain of the internal transactions which are images of the target segments. This concept permits widely varying external transaction formats that can conform to document design or the user's peculiar preferences, while at the same time conserving a relatively stable update programme. The Edit/Update Control table, describes on the following pages, are generated from the dictionary whenever there are changes in the transaction formats, data editing rules, etc.

EDIT CONTROL



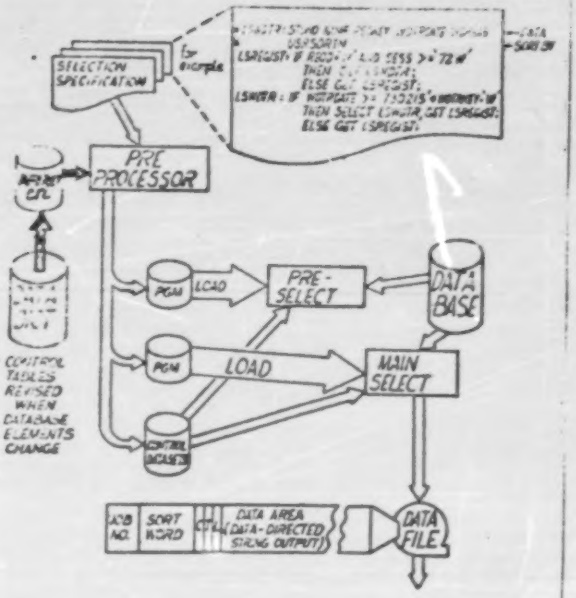
Control flow in the Edit Control Tables begins with the transaction code located in the INPUT TRANS TABLE, which points to the INPUT TRANS FLD TABLE for the fields. Each field refers to a TEMP STORAGE/VER TABLE entry, which contains the editing rules (table lookup, special routine, or RANGE TABLE /INDEX-SRCH TABLE lookup). Once the entire external transaction has been thus unpacked, any internal transactions signalled are then constructed, with O/P TR TABLE and O/P TR FLD TABLE referring to the same TEMP STORAGE table.

UPDATE CONTROL


TARGET SEGMENT NO. TRANSLATION
SEG PTR
SEGMENT TABLES
SEG. NAME
DEPTH
KEY FLD NAME
PAR PTR
SEG #
SPEC RTN
MAST OUT
FLD PTR
# FLDS
USED IN SSA CONSTRUCT
SPECIAL TRANSACTION ROUTINES (E.G. LOGICAL RELATIONSHIPS, UPDATE OF SUMMARY COUNTERS, ETC)
MASTER PRINT OUTPUT ROUTINES
FIELD TABLE
TYPE
START
LENGTH
DATA TYPE
ROOT REF #
ROOT FLAG MAINTENANCE ROUTINES
SPECIAL DEFAULT ROUTINES

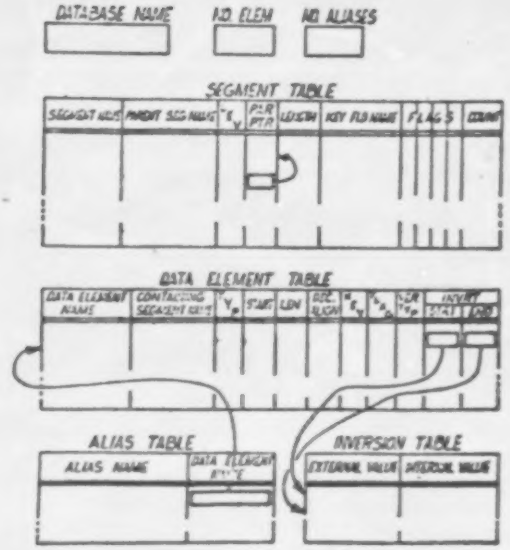
In the Update Control Table, flow begins with the internal trans. # translated to an entry in the SEGMENT TABLES, which contain all the information for construction of the SSA chain required for the update IMS call. The segment table refers to the FIELD TABLE for field-by-field replacement of the I/O area, or to implement default values as required. References are also made to special routines, root flag maintenance, and resultant master printouts.

INFORMATION RETRIEVAL



The Information Retrieval process is shown on the chart. The user codes a selection specification in a special language (a variant of PL/I), which is processed to compile two programmes and some control datasets. This pre-processor operates under control of the INFO.RETR.CTL dataset generated from DATADICT. The two compiled programmes are a pre-select and a main select, and the final output is a data file with the data string in DATA-DIRECTED STREAM format.

INFO. RETRIEVAL CONTROL



The control tables for the information retrieval pre-processor consist of (1) a SEGMENT TABLE, containing the information necessary to construct SSA's, and (2) a DATA ELEMENT TABLE, with all information for determining conversions, etc. An ALIAS TABLE is also provided to permit users to refer to different names for the same data element. The INVERSION TABLE refers to differences (apart from simple conversion) in the external and internal codes for an element.

RELATED UTILITIES

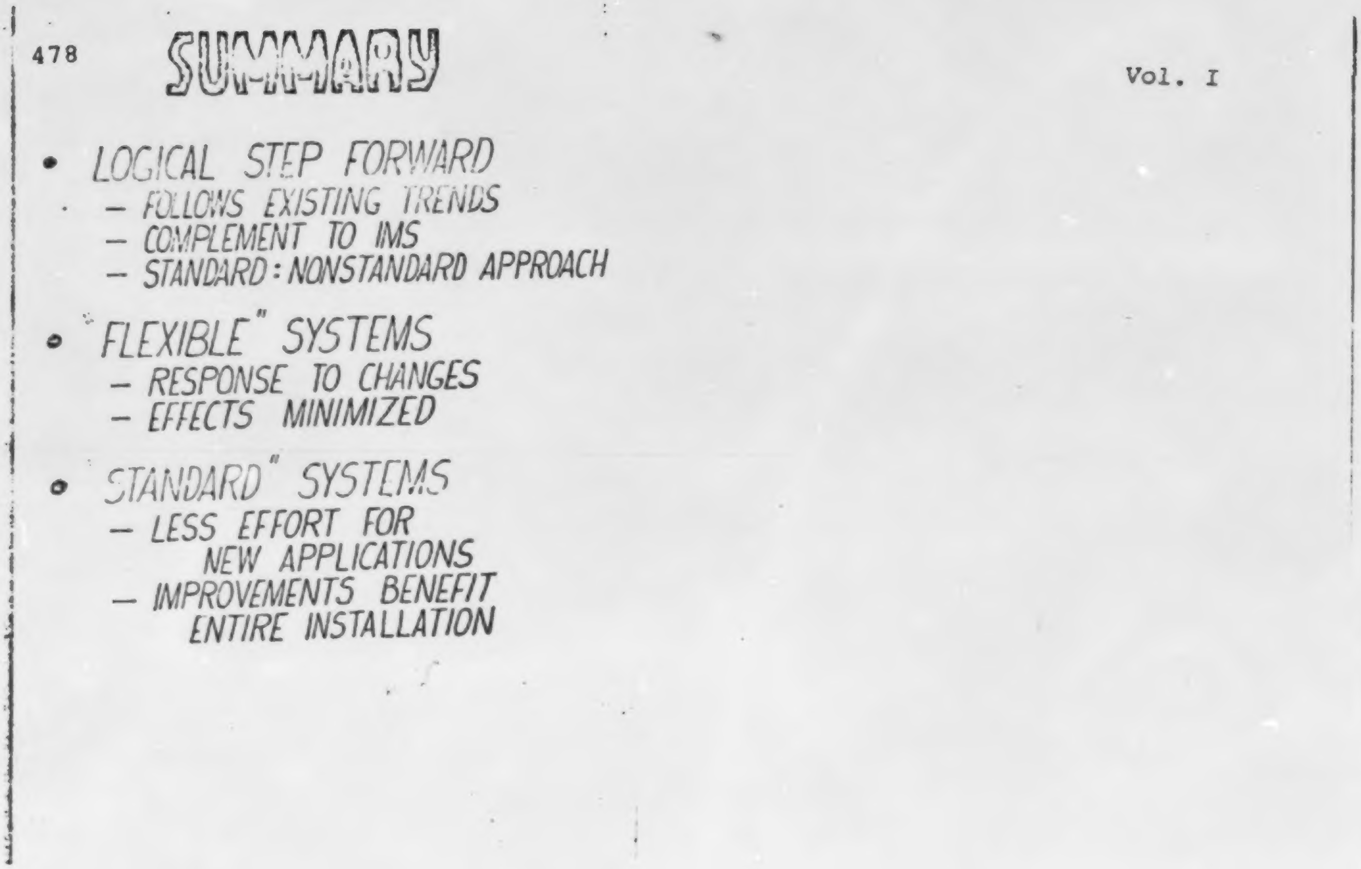
- GENERAL DATABASE LOAD
  - LOAD ANY DATABASE
  - PERMITS TEST DATABASE BEFORE SYSTEM PGMS READY
  - CURRENTLY REQUIRES EBCDIC CODING OF PACKED OR BIT DATA
  - FUTURE-TEST DATA GENERATOR
- GENERAL DATABASE DUMP
  - DUMP ANY DATABASE, IN CHARACTER & HEXADECIMAL REPRESENTATION
- GENERAL DATABASE CONSOLE ACCESS
  - ALLOWS ON-HAND TRACING OF DATABASE PROBLEMS

To assist the system designer further, a number of related utility programmes were developed along with DATADICT: (1) a general database load programme, to permit a test database to be generated before the actual system programmes are implemented; (2) a general database dump programme, for batch debugging; and (3) a general teleprocessing database dump, for on-line debugging.

FUTURE

- REPORT GENERATION
  - EXTENSION OF INFORMATION RETRIEVAL PROCESS
  - SIMPLE REPORTS: TITLES, COLUMNAR HEADINGS, ACCUMULATION OF FIELDS, ETC.
  - COMPLEX REPORTS
  - OTHER MEDIA
- CONVERSION AIDS
  - AUTOMATIC PRODUCTION OF CONVERSION PROGRAMMES
  - DATA CONTENTS ANALYSIS
- INTEGRATION WITH UNITABLE SYSTEM
- TELEPROCESSING SYSTEM EDIT

In future versions of DATADICT, several improvements and extensions are foreseen. The largest area for improvement is that of REPORT GENERATION, although this may be largely superseded by GIS/VS. The DATADICT system could be of immense value in database conversions, by automatic generation of conversion programmes and periodic data contents analysis. A closer integration with the UNITABLE system, also used in the generalized edit, is to be done, as is the development of a general protocol for TP Edit in direct data entry systems.



478

SUMMARY

- LOGICAL STEP FORWARD
  - FOLLOWS EXISTING TRENDS
  - COMPLEMENT TO IMS
  - STANDARD : NONSTANDARD APPROACH
- "FLEXIBLE" SYSTEMS
  - RESPONSE TO CHANGES
  - EFFECTS MINIMIZED
- "STANDARD" SYSTEMS
  - LESS EFFORT FOR NEW APPLICATIONS
  - IMPROVEMENTS BENEFIT ENTIRE INSTALLATION

In summary, a data dictionary system is a logical "next step" in the existing trends in data processing and information systems. It operates as a complement to IMS, filling in the holes and increasing the utility of DB-oriented systems. By identifying the standard areas and developing a common solution, the area of concern can be reduced considerably to focus on those elements that are non-standard. It makes it feasible to produce "flexible" systems that can respond quickly and effectively to changes in the environment, with the minimum effect in time and effort; and at the same time these systems are "standard" and uniform throughout the shop, with correspondingly less effort to develop new applications. Any improvements in the DATADICT system will benefit all applications in the installation by increased productivity.

DATA BASE ADMINISTRATION, IMS, AND THE MASTER DATA BASE

William B. McKelvey
Director
Administrative Systems Development
Ohio State University

DATA BASE ADMINISTRATION, IMS, AND THE MASTER DATA BASE

A. Problem for Tomorrow

We are emphasizing the development of operational data bases today and tomorrow but, in the predictable future we will have these data bases built. What then?

The problem is: how to utilize these operational data bases to support comprehensive reporting in a timely and efficient manner? The key to the answer, I propose, is the many and varied codes translation files, mapping files, "look-up" tables, or whatever you call them. We should take all of these mapping files as I will call them and with them build a data base with date as the root segment.

This data base gives us the "keys" by which we can query the operational data bases, it serves as a controlling method for the creation, update, and use of these mapping files, and it provides a very useful description of the University and its environment. The data Base Administrator has the day-to-day knowledge of the existing mapping files and also the ability to create an IMS structure capable of supporting the concept. The Data Base Administrator must provide the key leadership in developing this data base I call the Master Data Base.

B. Current State of Administrative Data Processing At The Ohio State University

The organization called University Systems is responsible for hardware operation, production systems operation, and administrative systems development. Beneath the Director of University Systems are the organizations of:

- Computer Operations
- Application Systems Operations
- Data Preparation
- Administrative Systems Development
- Systems Programming
- Customer and Support Services
- Library (Programs & Documentation)

Within Administrative Systems Development are the functions of systems analysis, programming, Data Base Coordination, and Data Base Administration.

Administrative Systems Development has the responsibility for systems planning (long and short range), systems development, and systems maintenance. The current long-range plan for Administrative Systems Development includes the building of five large IMS data bases devoted to Students, Personnel, Curricula, Facilities, and Financial information. Currently, we have built the Student Data Base and the Personnel Data Base. Plans for the next calendar year include defining the requirements for the Curricula and Financial Data Bases. In addition to the concept of the five major data bases, we have been increasingly encouraging the separation from programs and consolidation of mapping files under the control of the Data Base Coordinator. We presently have identified, found, and/or created over 100 of these mapping files ranging in size from several entries to over a 1,000 entries.

C. Areas of Special Attention When Using IMS

1. Planning the structure of the data base

The impact of changing the format of an existing segment is significant. It is easier to add new segments. The flexibility and programming ease of using IMS is achieved when the need for new data or a change in format of existing data is performed by adding a new segment to the data base rather than changing an existing segment. If a segment is added, only those existing programs needing the data will have to be changed. But, if an existing segment is modified every program that accesses that segment may have to be changed. This situation is especially true if there is not enough filler at the end of the segment to hold all of the new data. In the future with the variable length segment capability the criticalness of available existing space in the segment will be eliminated. However, the impact of changing segment format will only be slightly reduced.

2. Size of segments

Segments should tend to be larger rather than smaller. Small segments usually mean many segments and the pointer overhead and accessing overhead can be very costly especially in the sequential processing mode. Since IMS requires a four byte pointer for every segment, a large data base can have a significant overhead in size. A large number of segments very definitely will affect processing time especially in the batch mode. On-line processing is also affected by a large number of segments per root segment if there is a need to access many of them at one time.

3. Levels of hierarchical structure

Minimize the number of levels within the data base since the levels add to the access overhead. The lowest levels should not have to be frequently accessed especially in the batch sequential mode. Faster processing is achieved by developing a long, shallow data base structure rather than a short, deep structure. Accessing lower level segments in a deep structure can incur more processing overhead than total production logic execution. A long shallow structure also makes adding segments easier.

The most significant disadvantage to the IMS hierarchical data base structure is the inability to have more than one or two access keys. This current deficiency aggravates the problem of deeply structured hierarchical data bases. IBM has announced the availability under IMS/VS of expanded secondary index capability that will allow multiple indexes other than the primary root segment. With this capability batch processing can be significantly quicker by making the search argument a secondary index.

4. Sequential Processing

Careful consideration should be made of the cost of processing an IMS hierarchical data base in a sequential mode. The trade-off of creating a "snapshot" of the data base for sequential processing (especially for reports) as opposed to running against the IMS data base can frequently show the "snapshot" method is efficient. The problem stems basically from the inability to have more than one main root and more than one or two secondary root segments. We are encouraged and hopeful that IMS/VS with multiple secondary indexes will significantly resolve this particular problem.

D. Control of Mapping Files

Since the early days of stored program computers the separation of data and programs has continually but gradually widened. We, as an industry, have not achieved the degree of real flexibility in our programs that we say we have. We still allow and in some cases encourage systems and programs to have imbedded "lookup" or mapping tables of data for translation purposes. Each one of these tables insures that the program code will have to be changed, recompiled, and in some cases the logic changed to accommodate a change in a code or an addition to the table. Although it is currently very impractical to remove every dependency upon data values from programs, it is necessary especially in an integrated data base environment to remove as much dependency upon data values from programs as possible.

Some of the reasons for moving quickly and decisively in this area are obvious - Minimize program changes

- Increase system and program compatability
- Reduce data incompatibility and allow increased comparison of data
- Reduce the number of mapping files or tables.

Some of the not so obvious advantages are:

- Increased ability to compare and report among major units of the organization
- Increase the usefulness of the mapping files themselves. They are a vast source of information.
- Increased potential management control of university resources
- Minimize the changes to the structure of the data bases
- Minimize the concern of programmers and analysts with code structures and meanings, thus reducing program maintenance.

For these reasons we are removing as many as possible imbedded mapping tables from existing programs and designing new systems without these mapping tables imbedded. The job of controlling the creation, maintenance, and deletion of mapping files rests with the Data Base Coordinator. Along with his duties of data element control he is the organization's contact for all outside users concerning these mapping files. The split of data base administration functions between the Data Base Administrator and Data Base Coordinator allows more emphasis to be placed in both major areas - data element control - data base structure, performance, and development.

At Ohio State the functions supporting data base administration are:

The Systems Programmer

1. Install new versions of IMS as issued.
2. Make recommended changes to the current IMS version as received.
3. Solve the more subtle IMS systems problems that are encountered in the programming or operations areas.

The Data Base Administrator

1. Interact as an IMS consultant in assisting systems analysts and programmers in defining specifications, answering questions, etc.
2. Design and implement physical and logical data bases.
3. Prepare Program Specification Blocks (PSB's) that are the interfaces between application programs and data bases.
4. Create and maintain test data bases and support programmers in the use of these data bases.
5. Analyze data base usage and tune data base parameters for improved performance.
6. Assist in establishing data base backup and recovery procedures.
7. Train new staff members when required regarding IMS programs.
8. Construct, as needed, Data Definition Tables (DDT's) required by GIS to query the various data bases.
9. Project data base growth and recommend reorganization cycles.

The Data Base Coordinator

1. Mapping File coordination and development.
2. Building the Master Data Base.
3. Test and develop user orientated languages.
4. Data Element Dictionary coordination.
5. Coordinate user access to data bases.
6. Generate programs to test reliability of data.
7. Interact with users to insure consistent input data.
8. Educate output users to data base problems and pitfalls for easier solutions.
9. Advise users on availability of data and data release controls.

10. Coordinate data base privacy.

E. Problems of the Future

1. How to Interrelate Five Large IMS Data Bases?

Since I feel we have all the ingredients to build and operate all five major data bases, the problem today is to determine how to get more than operational benefits from them tomorrow. How can we get planning and control information out of them? Using our philosophy of building from the bottom up - How can we interrelate two or more data bases efficiently (quickly and economically) to produce information not directly computable before? I believe I can characterize this concept by verbalizing this sample request - "Provide the Provost of the university a summary of all the resources assigned or used by a degree program." Maybe a more attainable goal would be to: "Summarize all resources used or assigned to a given department".

The problems of identification of the data, the extraction of the data from each data base and the summarization would be a very time consuming and costly operation today since we have few capabilities (such as selection criteria and programs to support efficient large queries) to support this request. This is a very simple request from a logical standpoint. We currently keep quite a bit of the operational data by department.

A more complex request would be to: "Identify the costs of research efforts in the college of Administrative Science".

How could the Master Data Base be used for satisfying this proposed request?

a. The Master Data Base is queried to determine the availability of the data desired.

b. A list is created of the data needed from each operational data base once the desired data is found.

c. This list is passed against each operational data base as required and the data extracted into "save" files.

d. The save files are either processed through batch programs or through generalized on-line processing modules to interrelate and format the data.

Since the inception of data base management systems and the integration of data processing systems the ability to provide data or information to satisfy an unanticipated request with minimal effort has been a goal. One of the main reasons for not being able to provide this kind of service is that we have not been able to describe the factors which define the information control structure of the organization.

The sum of the mapping files of the organization is this control structure. If taken together it is possible with few exceptions to build a description of the university - a skeletal representation of the structure of the information system control parameters.

If these mapping files were used to build a sixth data base using date as the root key, then the data can be used for operational processing, for gathering information from itself, and for the selection criteria and control for unanticipated requests. This sixth data base, called the Master Data Base, will have a hierarchical structure as shown in diagram (A). The root segment must be keyed on date since date is the only common denominator among this "conglomerate" of data. Each of the second level segments represents a major mapping file occurring once per root segment. This allows systems which function on differing calendars to share common mapping data. It is also anticipated that there could be supplemental indicators within the mapping file segment indicating which systems can use the mapping data.

As integrated data bases expand, archival problems also increase. In the area of student transcript data this is especially important. There are several ways to resolve the issues of archiving student transcript data but most are compounded by the changing coding structures of mapping files. As the transcript is archived in computer processable form, the mapping files which are used to translate the coded information must also be stored. Only through some identification of the mapping files which are appropriate for the archived data can the correct data be accurately retrieved. The Master Data Base will serve this function also.

2. The Need for an Integrated Data Element Dictionary

Since the Master Data Base is initially envisioned to incorporate the data from mapping files, all of the needed data will not be contained within it. There is a need to have a data element dictionary type data base to define all of the available data stored in operational data bases.

This is especially true of the information about the individual data elements in each data base. There is a need to have a data element dictionary that is machine processable to support the operational and Master Data Bases. The data element dictionary serves the information query process much like a thesarus. With capability to "browse" through the data element dictionary the inquirer can "pick and choose" the data he wants so that the query can be developed to extract and summarize the data. The data element dictionary also serves the programmer through standard definitions, data base structures, and common edit and processing rules for each data element. The data element dictionary supports the control of data base design and expansion, control of program and system development, and management control of both the user and data processing organizations by collecting detail information about each data element in a data base and providing easy access to it. Management can for the first time see what the data is and the manner in which it is used.

At this time there exists a real need for a truly data element dictionary support function within IMS. The building by users if data element dictionary systems aggravates the processing problem of large integrated data bases by forcing the addition of another data base interface. Until support for a data element dictionary system is provided by IMS, the Master Data Base cannot provide its total service.

3. The Need for an Efficient Integrated Query System

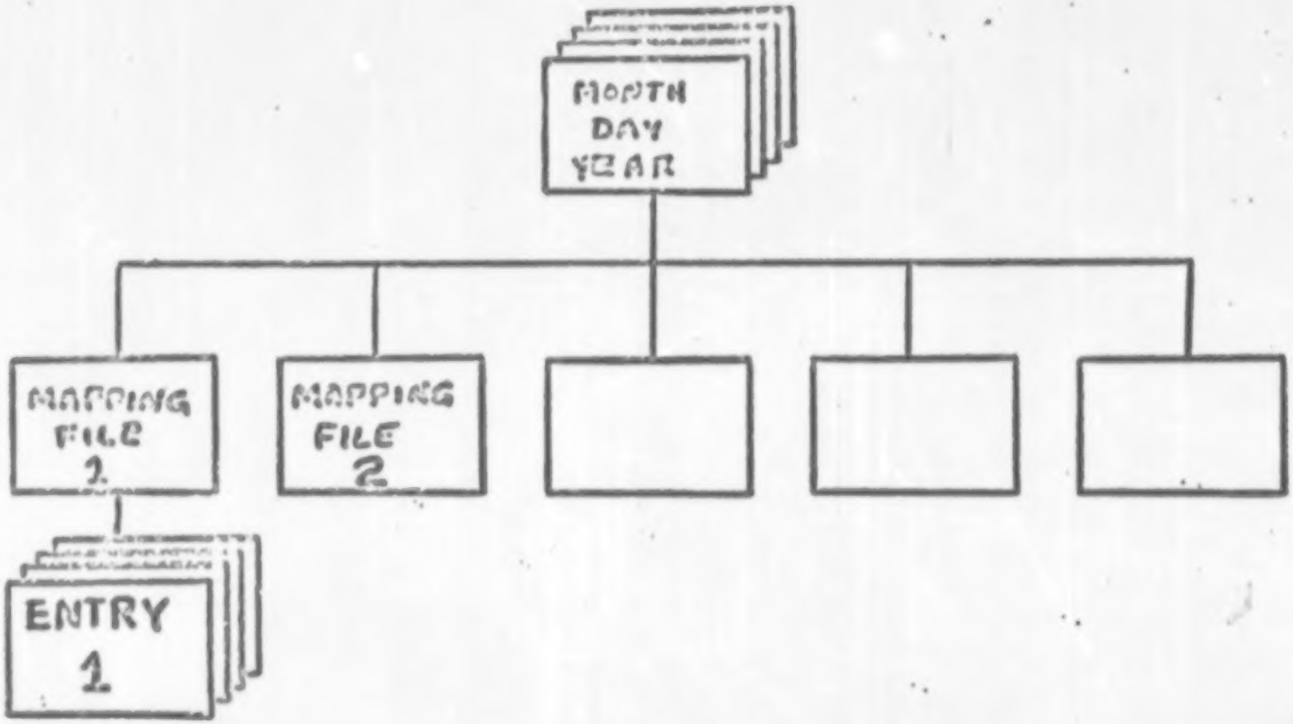
As can be deduced from my sample of how the process of using the Master Data Base and the operational data bases works, the need for a query system is another supplemental issue that needs careful study. Is there an existing query system or one in development that can provide the necessary command structure and flexibility to allow efficient processing in a limited use environment? This question must be pursued along with the data element dictionary support.

4. Cooperation

The role of IMS or other data base management system in supporting a comprehensive college or university information processing system has not been investigated thoroughly by the developers of these systems or by the organizations using them. When you look beyond the operational data base development effort, there is a large void of real planning and support from both the vendor area and the user community. With operational data bases being built and used today, it is but two to four years before we will need real support in the areas I have mentioned - Master Data Base, Data Element Dictionary support, and Query language support. Since the vendors do not have the total knowledge of the data base management system future needs, especially within colleges and universities, we, the, user must have a sound plan for future comprehensive systems development. We must communicate our requirements to the vendors and be willing to cooperate with them in designing the kind of data base management system support services we need. If we are to ever get the needed support it must come from the vendors. Developing these capabilities by individual colleges or 'universities' would be too expensive and would not provide for easy sharing. I feel also that software alone cannot satisfy the entire need. Hardware must play an integral role in supporting advanced comprehensive data base management.

5. Data Base Administration

The problems and challenges of Data Base Administration are ever growing. As I have mentioned there are no answers today and may not be in the near future unless the function of Data Base Administration is made an integral part of the total information processing effort. Data Base Administration must assume this role and lead in the areas discussed today without neglecting its day-to-day obligations. The era of Data Base Administration is coming. Unless it prepares and plans carefully today, it will not be ready to assume its total responsibility in the future. If integrated data base information systems are to succeed, we must insure Data Base Administration has the tools it needs to do its job and we must prepare it to perform its vital functions.



MASTER DATA BASE

DIAGRAM A

SUMMARY OF SYSTEM 2000 DATA BASE MANAGEMENT SYSTEM EVALUATION PROJECT

Mark E. Ragel
Systems Coordinator
Southeast Regional Data Center
Florida International University

Martha A. Fields
Systems & Programming Manager
State University System of Florida

DATA MANAGEMENT IN THE STATE UNIVERSITY SYSTEM OF FLORIDA

These first few minutes we will describe the framework and direction in which Management Information Systems in the State University System of Florida is moving. We began several months ago thinking data base and data base management systems. Now we have broadened our prospective to Data Administration and Data Management. I'll describe some of the processes leading to our current conceptions of these philosophies in our environment.

The collection of information to support the decision-making process of management has been the goal within the State University System of Florida for a number of years. This objective has many facets and has been approached from a number of directions, but the basic goal has remained that of defining and implementing a set of policies and procedures which interact with data so as to impact management decisions in a timely manner.

At the beginning of Management Information in the State University System, several assumptions were made which are still valid today:

1. The best data for an MIS is found in the systems supporting the operational requirements of the institutions.

2. The most accurate data is collected at its source by the person most familiar with the operation.

3. The information needed to manage the State University System is a subset of that needed to support the institutions.

Efforts to date have been somewhat frustrated by a lack of tools.

Our approach to date has involved three major areas: Regional Data Centers, UNIFTRAN, and Data Tapes. Our first approach concerned computer resources. Access to comparable levels of computer hardware support at each institution was not uniformly available. In 1969, four Regional Data Centers were formed to support the computer requirements of the nine state universities. The formation of the Regional Data Centers was a major step in solving access to hardware resources.

A second problem area was the wide range of sophistication of systems at institutions which support the administrative processes. The UNIFTRAN effort was begun to meet this need. Standing for Uniform Transactions, the term represents our standard systems development, where all nine institutions use the same systems for basic operational areas such as Personnel and Payroll, Accounts Receivable, and Purchasing.

Thirdly, the Board of Regents office collects certain information from the universities on a quarterly basis. There are six different files which represent a snapshot of the institutional files. The information is used to fill reporting and information requests of the Board of Regents.

As we reviewed our progress to date, we felt several problem areas remain. For example:

1. Operational systems lack a commonly defined base of data. We found this leads to redundancies and inconsistencies when certain data is viewed system-wide.
2. We recognize an inability to communicate effectively across systems where logical relationships exist.
3. We recognize an inability to react quickly to increased demands for ad hoc information requests.

A Data Base Management System can provide the logical structure to cross systems and to provide faster answers to ad hoc requests, but there is a broader perspective to the areas we wish to improve. We feel the Data Management approach addresses our needs and I will discuss this briefly in light of what we feel Data Management is and how it impacts the State University System of Florida.

Data Management is the combination of software and personnel resources to implement policies and philosophies relative to viewing the status of the organization through data. Data Management has emerged to address problems of data redundancy, inconsistency, security, accessibility, and control. Data in this environment is viewed as a resource of the entire organization and is processed with organizational goals in mind.

The software tools of Data Management are emerging as a proven technology within reach of the State University System. For our environment the software will include a Data Dictionary System, an Input/Output Controller, and a Data Base Management System. The combination of these tools in a Data Management environment provide a disciplined method of managing information. Data is available in a form which is more beneficial to management needs in light of operational responsibilities. Data is created, collected, controlled, and discarded with organizational needs in mind.

The area of personnel utilizing this software has come to be known as Data Administration. The function of Data Administration will exist and operate at the Regents Office, at the Institutions, and at the Regional Data Centers. The global functions of Data Administration are coordination of data definition, control of access to the data, training and consulting, procurement and maintenance of the software, and the definition of physical data structures.

At the Regents Office, Data Administration is in the Management Information Office and will address data definition and control for data bases to support the BOR information needs, systems consulting for UNIFTRAN and for general applications, training in data management techniques, and coordination of software systems maintenance.

At the Regional Data Centers, Data Administration will address software systems performance and technical consulting. At this level, Data Administration will interface the data management software into the operating environment of the Centers and provide access to the software and data for authorized users.

Data Administration at the Institutions will address coordination of data definition and control of the instaitutional data base. Viewing data as a resource of the institution as a whole requires an agent to execute the institutional policies on availability and access. Sharing data resources and structuring data in an inter-related form provides a more viable means with which to support the organization, but definitions must be clearly stated and resolved among all users to realize the benefits of consistency and lack of redundancy. The Data Administration function at the institution will use data management tools to coordinate the data resources in light of the institutional management policies.

The interaction of personnel and software in the data management environment will provide the State University System with more timely and accurate information.

One step we've taken is the implementation of a policy which channels all requests for data through our office. Monitoring the information presently used will enable us to structure the data bases for this office more effectively. The processing of these requests has enabled us to answer institutional questions and has enabled us to resolve some discrepancies in our data definitions. As this policy continues we will better understand the needs of the Board of Regents and the Institutions and be able to provide true service.

We are presently writing specifications for a dictionary system. We will use the dictionary to inventory the data requirements of the Regents Office. We will catalogue validation rules and controls for access and security in addition to clear definitions for each item.

We have acquired a Data Base Management System -- System 2000 -- and have the basic system installed on three of the four Regional Data Centers.

Our Data Base Management System selection procedure involved several phases. These included the decision to survey, the development of specifications, a survey of available systems, the in-house trial of System 2000, the evaluation, and the proposal to acquire the system.

Mr. Mark Ragel will now describe the activities which comprised the in-house trial of System 2000.

INTRODUCTION

Nolan (1973) describes three significant disadvantages of traditional file techniques as: 1) Files and records have tended to become redundant; 2) The traditional approach undercuts or aborts the advances of computer technology; 3) The traditional approach obstructs upper management's growing demand for applications that require a data base. Schubert (1972) echoes a growing sentiment in the data processing industry when he says, "clearly the need exists for a data base management system to provide: control of data structure; comprehensive language for data manipulation; adequate execution performance; and system integrity." Sibley and Merten (1972) indicate that a consideration in the selection of a data base management system (DBMS) should be transferability to move the emerging organizational data base from the computing system upon which it was developed to other computing systems as may be required.

The present study investigates the performance of System 2000 (S2K) (a generalized DBMS marketed by MRI Systems Corp, Austin, Texas) in a multi-vendor (IBM/UNIVAC) University computing environment.

OBJECTIVES

1. To evaluate S2K in an actual university computing environment; to determine, in fact, the capabilities and benefits that would result from acquisition of this DBMS.
2. To evaluate the performance of S2K in batch mode procedural language (COBOL) systems.
3. To determine the true degree of transferability of applications developed under S2K.
4. To ascertain the "real world" benefits of the extended retrieval capabilities of S2K.

EQUIPMENT

Northwest Regional Data Center (NWRDC)
Tallahassee, Florida

Machine: IBM 360/50

Operating System: MFT/OS - HASP (Rel. 21.7)

Data Base Management System: System 2000 (IBM version Rel 2.4)

Southeast Regional Data Center (SERDAC)
Miami, Florida

Machine: Univac 1106

Operating System: Exec 8 (Rel. 31.159)

Data Base Management System: System 2000 (Univac version Rel 2.5)

NOTE: For a feature comparison of the respective versions of System 2000, see Editor's Note.

APPROACH

To determine that S2K would provide adequate execution performance in typical university computing environments and to ascertain the transferability, across vendor lines, of systems developed utilizing S2K, both the Northwest (NWRDC) and Southeast (SERDAC) Regional Data Centers were designated as test sites. Figure 1 shows a flow of major activities.

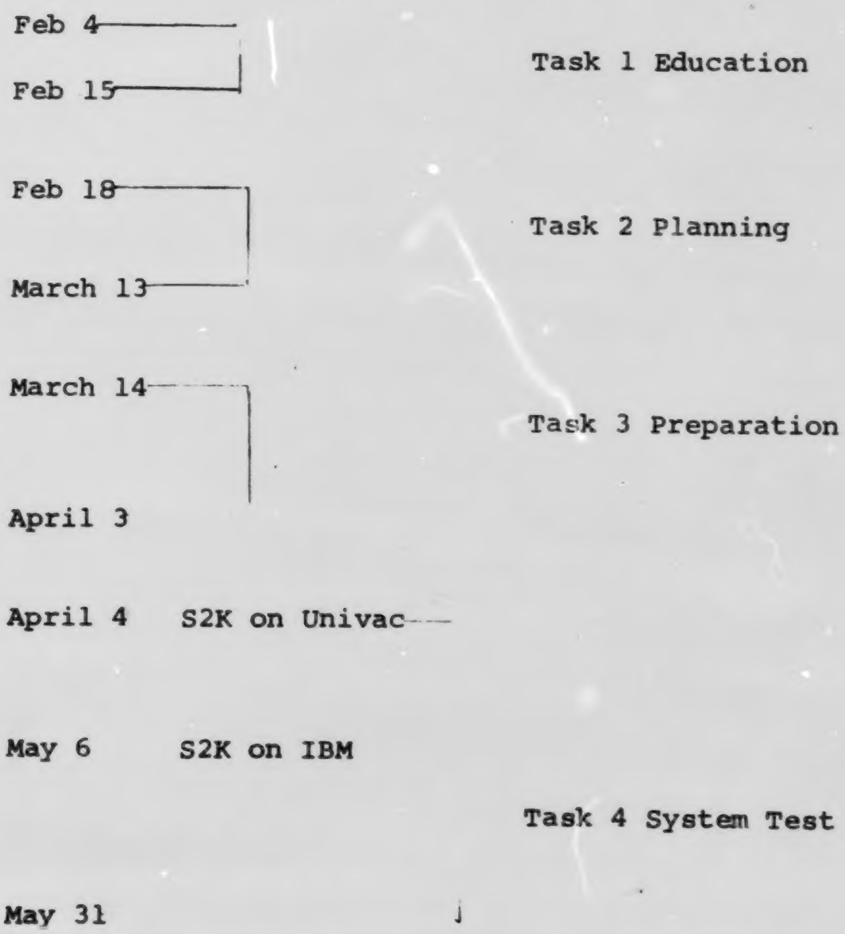


Figure 1

WORK PLAN

TASK 1 EDUCATION

The formal education phase of the study was comprised of two courses offered at the MRI Systems Corporation's educational center in Austin, Texas. The first course, "Basic System 2000", covered natural language, data base creation, and data base access through "Immediate Access" and "Queue Access". The second course, "Procedural Language" covered the COBOL and FORTRAN interface with System 2000.

TASK 2 PLANNING

The planning activity of this phase included consultations and system reviews covering the range of activities that are served by the Office of Institutional Planning and Budget at FSU, the FSU Computing Center, the MIS office of the BOR, NWRDC and SERDAC. As a result of these conferences the following decisions were reached:

1. To attempt implementation of three data bases as follows:
    - * 'Authposbse' - a data base composed of the logically related information items contained in the currently discrete B.O.R. reporting files, Authorized Position and Faculty Activity.

- 'Stucsebse' - a data base designed in accordance with the Uniftran* student information inventory.
- 'Paymstr' - a generalized payroll data base.

2. To utilize the authorized position update function as the basis of the data base vs. conventional file, batch mode procedural language (COBOL) system comparison; this decision was reached with the full knowledge that System 2000's forte is retrieval, not update.
3. To acquire a level of proficiency with the natural languages that would facilitate the presentation of meaningful demonstrations of the retrieval capabilities of S2K.

4. To utilize the Univac 1106 at SERDAC as the primary facility for developing experimental systems and to transfer systems to the IBM 360/50 at NWRDC to determine S2K's compatability across hardware vendors.

---

* A Uniform - Transaction system of the State University System of Florida

TASK 3 **PREPARATION**

This phase of activity encompassed: an intensive system study of each area to be implemented as a data base; the design and coding of the respective data bases; the design and partial coding of the COBOL programs required to load the respective data bases; the design of the COBOL program required to perform the authorized position, transaction update function against the Authposbse data base.

TASK 4 **SYSTEM TEST**

On April 3, 1974, S2K was installed on the Univac System at SERDAC; at 9:00 a.m. the MRI Technical Representative began the first of eight jobs that would install the DBMS. By 12:00 noon the installation had been certified and the MRI representative was preparing to leave.

Testing began immediately utilizing the S2K natural language for construction of the test data bases. Similar to writing a COBOL file description the natural language construction of an S2K data base is a quite straight forward task, with simple syntax and use of 'PICTURE' clauses. Coding was completed and testing began on the COBOL programs that would be employed to load the data bases from conventional files. Although it was deemed most expedient, in this instance, to use a procedure language interface (PLI) program to load the test data bases, S2K's facility to load data bases through the natural language would be appropriate for other situations. The S2K COBOL PLI utilizes a pre-compiler to scan COBOL source code for S2K statements. (All S2K statements start in card column 7 rather than column 8 as used by conventional COBOL statements.) Based on an evaluation of the statement encountered, the pre-compiler either generates diagnostics, or inserts source code and passes the expanded source program to the standard COBOL compiler. The diagnostics generated by the pre-compiler are reasonably clear and specific.

Although one rather frequent diagnostic is somewhat obscure the interface to S2K through COBOL presented no problems of any consequence. In principle, the retrieval/update logic required to interface COBOL with an S2K data base is quite similar to COBOL index sequential retrieval/update logic, although, the additional concept of 'positioning' (within the data base hierarchy) must be understood to make good use of the DBMS facilities and to correctly predict the outcomes of many commands.

In the course of the testing conducted pursuant to this study, S2K proved itself to be a reliable and accurate package. There was only one 'Quirk' (having to do with the order in which certain data types were described in a schema) encountered in the course of the study; it seems worthy of note that the solution to the problem was provided in a most expeditious manner by the technical support staff of MRI.

On May 6, 1974, S2K was installed on the IBM system at NWRDC with the same relative ease experienced with the Univac installation.

At the time of the NWRDC installation, the 'Authposbse' data base was operational on the Univac system and the program to load the 'Stucsebse' data base had undergone compile error correction and was in the initial stages of testing. The decision was made to convert Authposbse and Stucsebse to the I.B.M. system. To acquire data regarding the transferability of batch mode S2K systems, the method used to effect the conversion of the data bases was to convert the COBOL PLI programs, and the raw input data, from Univac to IBM, and then proceed to load the data bases in the same manner as they had been loaded on the Univac system.

The difficulties encountered in transferring these systems from Univac to IBM were due exclusively to the incompatibilities in data recording techniques and COBOL compilers of the respective computing systems. Notwithstanding the considerable operating environment differences between the computers used in this study, it is apparent that from a user standpoint, S2K is virtually 100% compatible within the confines of the package.

At approximately the time of the S2K installation at NWRDC, arrangements were completed to demonstrate the conversational mode retrieval capabilities of S2K at the MIS-BOR* offices at 107 W. Gaines St., Tallahassee on May 15, 1974. An intensive effort was undertaken with members of the Board of Regents in the planning and preparation of a presentation, which culminated on May 15 with a demonstration of S2K's query language capability to Chancellor Mautz (State University System), Wm. Corbett (E.D.P. Division), members of their respective staffs

* Management Information Systems- Board of Regents

and representatives of various other agencies of the State of Florida. Between May 16 and May 31 BOR staff members, and the author, conducted in excess of 20 additional query language demonstrations of S2K at various locations throughout the State. Subsequent to the May 15 demonstration, coding was completed and testing begun on the COBOL PLI program required to perform the update function against the Authposbse data base.

RESULTS

The findings of this study, see Editor's Note, are described as, "advantages and disadvantages of S2K , as related to an idealized DBMS". (Everett, Brady and Wilkes, 1973)

DISCUSSION

From the standpoint of the information derived from this study it seems appropriate to respond to the "Six Most Important Questions in Evaluating a Data Base Management System" as defined by the First Cycle Results of the CAUSE*Delphi Project (WHITE, 1973)

QUESTION	ANSWER
1. Is the package operational now and has it been in a using environment for several years?	Yes, the first C.D.C. installation was in 1969; the first IBM installation was in 1972: the first Univac installation was in 1971; currently over 200 users.
2. Are there institutions similar to ours using the package?	Yes, at least 5 other Universities.
3. Can the package add, delete modify simultaneous with query and report question?	No, at present update functions require exclusive use of the data base; a revision in this area is said to be forthcoming.
4. Can appropriate backup copies of the data base, including the latest transactions, be generated in case a file is destroyed?	Yes, categorically, S2K has extensive backup capabilities.

---

* College and University System Exchange
Boulder, Colorado 80303

QUESTION	ANSWER
5. Is backup/restart available?	Yes.
6. Can the package retrieve data from two or more files within the same run?	Yes. In a PLI program up to eight discrete data bases may be accessed, with any four open at one time.

In the course of this study it was found that, although a proportionally large amount of time and effort is required to design and implement any efficient data base, the user oriented simplicity of S2K minimizes the learning curve that is attendant to this new technology. This is not to say that there are not some real costs associated with acquisition of the power of S2K. On the SERDAC system: Initial load times of 12,850,000 character data bases can exceed 3 C.P.U. hours; compared to a COBOL sequential file update program, an S2K COBOL PLI update program (in unoptimized form) can require 30% more C.P.U. time. (It is worthy of note that S2K run time improvements of up to 20:1 have been reported as a result of the application of optimizing techniques.) But, the extended capabilities of S2K appear to offer a solid base for the development of systems that can evolve, in an orderly fashion, as information demands and computing environments undergo continuing change; and the retrieval capabilities offer tools to applications technicians, and non-technical information consumers, that provide a level, and ease of access to information that has been heretofore unavailable.

SUMMARY

The primary findings of this study fall into four main categories:

1. Retrieval Capabilities.

   The Immediate Access, Queue Access, and Report Writer features of System 2000, represent a new generation of software that, by virtue of its simplicity, can provide even a relatively naive information consumer, a level of access to information that is simply unavailable via conventional systems.

2. Hardware Independence.

The user oriented, virtual 100% compatability of System 2000 across CDC, IBM and Univac computers, provides a capability for expeditiously developing uniform systems that would be totally unfeasible utilizing conventional system techniques.

3. Implementation.

It is apparent that by utilizing the facilities of System 2000, design and construction of a data base can be accomplished in a most expedient manner, and at the point the data base is operational, in many respects, response to information needs can be virtually instantaneous.

4. Maintenance.

With the type of data independence provided by System 2000 data bases, the inclusion of new data elements no longer necessitates revisions of previously implemented information systems.

As reported by Appleton (1973) "It is not often recognized, but of the three basic classes of data processing resources, hardware, software, and personnel, computer software is the least flexible, most constraining, most costly and most time consuming. The simple truth is that a data processing organization can only do what its computer software will allow it to do. As it turns out, the traditional approach to the development of information systems in a large part defined by the software available, rather than the hardware, personnel, business environment or development problem. It was not until System 2000 was encountered that an alternative to deterministic systems development became clear and,at the same time, practical." (For summary and conclusions of other available studies, see Editor's Note.

Acknowledgements

My special thanks to Hazel Adamik (MIS-BOR), Martha Fields (MIS-BOR), Pat Fisher (FSU Office of Institutional Planning and Budget), Don McEwen (FSU Computing Center, and John Thiel (NW Regional Data Center) for their time and good advice. Also my thanks to all of the excellent personnel at both NWRDC and SERDAC, especially operations, without whose cooperation and support this study would have been impossible.

EDITOR'S NOTE:

A lengthy appendix was submitted with this paper and is available in its original form from the CAUSE National Office. The topics covered are as follows:

Appendix A	Feature Comparison of IBM and Univac versions of System 2000
Appendix B	Advantages and Disadvantages of System 2000
Appendix C	System 2000 Statistical Capabilities
Appendix D	Project Planning and Management Information System Council of Great City Schools, Washington, D. C.
Appendix E	Data Management System recommendation from The Texas Water - Oriented Data Bank Committee
Appendix F	Program Information Control and Retrieval System (PICRS) Software Feasibility Study Final Report National Aeronautics and Space Administration, Houston, Texas

THE DATA DICTIONARY/DIRECTORY:
YOUR MOST IMPORTANT DATA BASE

Daniel S. Smith
Data Base Administrator
Indiana University

THE DATA DICTIONARY/DIRECTORY - YOUR MOST IMPORTANT DATA BASE

ABSTRACT

This paper discusses the uses and purposes of the Data Dictionary/Directory as a major component of the software necessary to support innovative systems. These uses and purposes apply particularly to multiple systems involving the integration of data or data processes in a manner which implies the application of sophisticated data management techniques. An examination is made of the role filled by the DD/D as a central software component of applied data management concepts interfacing with such other components as data base management systems, application programs, and operating systems; and with data base administrators, systems analysts, and "end users."

Ideas are presented in the paper about the content of your DD/D (your "data about data") including suggestions on how to efficiently collect and load that content beginning with your existing systems and files.

Although the DD/D is most widely recognized as an innovative tool for use in managing data bases, this paper illustrates how valuable the DD/D can be in managing existing files and using those files in planning for data base conversion.

BACKGROUND AND DEFINITION

In most institutions where innovative systems are being planned, designed, and developed, managers are recognizing data to be an institutional or "corporate" resource. That recognition represents a tremendous shift in understanding and emphasis about where data originates, what processes it undergoes, and how it is accessed and released (both internally and externally to the institution). Until data was regarded as a resource which belonged to the institution instead of a division or department, there was only minimal awareness that data (as a resource) required management, planning, and control similiar to that required by other resources. The real shock to college and university administrators, however, does not come so much from the discovery that data is an institutional resource, but that the data that presently exists in the various parts of the institution cannot be managed, let alone controlled. For example, many institutions find their data to be conflicting, inaccurate, superfluous, or wasteful. In addition, the documentation and coordination are often so poor that proper management and control are difficult, if not impossible, to add to the existing framework. The knowledge about what data exists is usually insufficient to allow the conversion to the resource idea through minor improvements.. This local predicament, heightened by increased reporting requirements from state and federal governments[1] and organizations such as WICHE/NCHEMS, necessitates that institutions develop innovative systems and upgrade their view and level of dealing with data. One important first step is the development of a Data Dictionary/Directory (DD/D).

The Data Dictionary/Directory in its present form is a tool designed to assist the institution in establishing data as a fully controlled and managed resource. The DD/D development into its present form paralleled the awakening of the "data as a rescoure" idea. In the first form of this evolutionary process the DD/D was typically a manually maintained list of elements by master file or a "file book." The file book did provide documentation for the files being used, but the systems were still fragmented so that major benefits were constrained by what could be done within a given system.

With emphasis on intersystem communications via use of the same or similiar data, the importance shifted from the system master file to the data element and the DD/D took on the data element dictionary (DED) form. Although the DD/D conceptually does not require computerization in any state, and it certainly is not limited to computer stored data, the overwhelming demand for it followed on the heels of the demand for computerized administrative systems especially with the mounting pressure for more advanced and more innovative systems. Therefore, it isn't surprising that most DD/Ds became computerized during the DED stage. The DED worked well and did provide the necessary coordination between people and data to promote better organization and progress toward the goals that necessitated data being viewed as a resource. The DED also served to allow more readily available statistics and to identify the potential problem areas of the fragmented systems. It did not, however, provide many benefits that would allow for the correction of problems through more responsive computer systems. What was needed was a machine interface between the DED and the processing programs, a "data directory." The data directory provided a method of translating the human data needs into machine usable form. Its importance is best understood from the aspect of change. As people's uses of data changed, the programs needed to respond. The directory provided the link to support a more dynamic interface between systems (or programs), data, and people. With the addition of the data directory the DED became the DED/D or usually the DD/D. The word "element" was dropped because the importance centered on data, not its form or a single attribute. It is this "final" form, the Data Dictionary/Directory or DD/D, its uses and implementation, that are discussed in the remainder of this paper.

Having established the need for a DD/D and discussed its evolution to the present state, some discussion of its definition is appropriate. The precise definition of a DD/D will vary slightly from institution to institution depending on the evolution within that institution, the objectives set forth, and the "state of the art" employed. However, there are fundamental ideas that should be considered before deciding on the final definition of what the DD/D is going to be at a particular institution. The DD/D is or at least can be (as the title states) a data base. However, a closer examination of definition criteria discloses that the DD/D is much more than a typical data base.

---

[1] For example, ability to adapt to the recent federal legislation on student information.

In the general sense, it is a depository of all of the information known about an institution's data. In other words, if one begins with the "corporate data" viewpoint and establishes rules for data utilization that conform to the institutional needs he could very methodically isolate the contents of the DD/D for that institution using a "top down" approach. This step-by-step establishment of rules along with isolation of the content should end with the "technicians" in the data processing department. The resulting repository of information about data called the DD/D must then service the needs of executives, middle managers, operational unit personnel, and data processing personnel. When the above discussion is restated in definitive form, the DD/D is "the conglomeration of the requirements of the data users throughout the institution to effectively describe and use that portion of the data necessary to perform each user's function."

An important first step in understanding the concept of a DD/D is the realization that the DD/D does not contain "data". It has no "content" in the common usage of the term. The DD/D contains only attributes about the data. The actual data or "content" is stored in or "contained" in another data base. As a matter of fact, one of the DD/D attributes about data tells in which of the other data bases the content for a particular element can be found. It follows that the description of the DD/D is described in itself as a data base, very similar to looking up the meaning of the word "Dictionary" in Websters.

The conceptual definition of a DD/D might then read: A tool in the form of a data base which contains all of the attributes about the description and use of the data by both people and computer programs in such a manner that more flexible and dynamic interactions are possible, and that management and control of data are facilitated.

IMPLEMENTATION

Whether or not this definition of a DD/D is completely acceptable, there remains the question of implementation. The implementation of the DD/D, like the definition, varies a lot from institution to institution. Thus, a completely "packaged" solution is not often available even though there are several good software packages on the market which can make the job easier.

The trend in implementation is to make the DD/D a special type of data base. As its title implies, this paper supports that trend. In its final form, the DD/D should be a data base or, in some more complex environments, a series of interconnected data bases. However, if the DD/D can be developed prior to the employment of data base technology, three or four related files can work wonders and serve later as the basis for the final development into a data base type DD/D.

The difficult part of implementation, though, is in determining 1) When to begin, and 2) How to fit the DD/D into the computing environment using available technology. The answer to the first question has been alluded to above and can be very simply stated—as early as possible! Although the final DD/D presented here implies the use of data base management systems (DBMS) and integrated data bases, many of the same benefits can be realized much earlier in the development cycle. In fact, some real advantages in planning for the DBMS environment can be gained by collecting and organizing what is known now about the data; in other words, by creating a DED.

DD/Ds in this DED state have proven to be valuable tools in organizing for a major development effort such as an MIS, especially where existing systems and data will be replaced. In fact, the DED offers valuable assistance in controlling existing systems in a more coordinated manner. The knowledge that duplication and disagreement exist at least places the administrator in a position to select the better data for the question at hand or properly qualify the answer.

It is easiest to begin as soon as the DBMS is selected and installed. However, beginning at this point will involve considerable overhead cost for the DBMS while the DD/D input is being collected and the "interfaces" determined. Therefore, it may be more practical to begin with a preliminary DD/D used to describe all data, programs, etc. existing prior to the DBMS installation. This preliminary DD/D is much less complex[2] than the final DD/D but is adequate to record what information is known about existing data. The preliminary DD/D can be implemented using a few "related" indexed sequential files to inexpensively provide descriptive information for planning purposes and later on for conversion purposes. This preliminary DD/D also tends to separate what is known about data from what needs to be known later. This separation allows utilization of the most knowledgeable personnel in each system to quickly record the information for the preliminary DD/D. In this manner, the DD/D can be inexpensively and quickly prepared for use as a development tool. Even if a DBMS and several data bases already exist this same approach can be used; however, the preliminary DD/D will necessarily be somewhat more complex. In either case, the DD/D can be developed with a minimum amount of resource investment distributed over the maximum number of people.

The answer to the second question on implementation (how to fit the DD/D into the computing environment using available technology?) is not as simply stated as the first. It has been established that for the most beneficial effect, the DD/D should be a data base. It follows that for most institutions where a packaged DBMS is used to handle the content oriented data bases, the DD/D should be a data base which operates under that same DBMS. Once the DD/D is created, it should serve as the "library" for defining all data bases to the DBMS.

---

[2] Sample input forms at Indiana University are included in Appendix A.

Unfortunately, the most widely used DBMS packages rely on a data base descriptor file or "table" instead of a DD/D as suggested here and as a result perform the desired functions in a limited, inflexible manner. Therefore, many of the benefits which should be available from the DD/D are only partially realized at best.

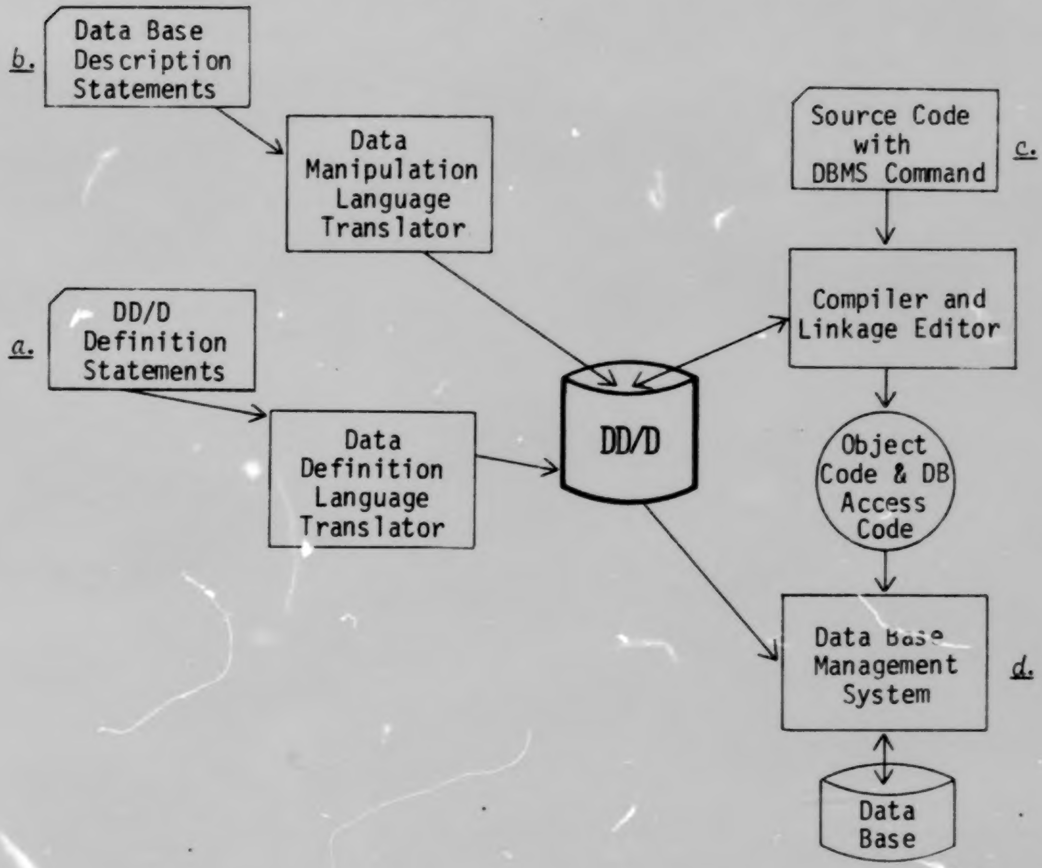
Since the DD/D, as defined in this paper, is not recognized by the typical DBMS an extra step is normally used to extract from the DD/D that information needed to build the descriptor tables for the DBMS under programmed control.

The two other steps missing from or weakly implemented in the DBMS-DD/D combination are at the program source level and the program execution level. These steps are needed to insure that the program can be kept as free as possible from the descriptive information about the data it references in order to allow maximum flexibility to meet changing requirements. A reasonable goal is that the program source code need know only the name or number of each data element used and whether that element is a member of a repeating group or non-repeating group in the data base structure. The program should not need to know the physical data base in which the element resides or the version of the DBMS being used. In fact, theoretically, the program should not be required to know even which DBMS package is being used. The interchangeability of DBMS versions or complete packages is handled by the addition of a Dynamic DBMS Interface Module.

The operating environment of the DD/D-DBMS combination is conceptually depicted in Figure 1. Since this conceptual approach implies an integrated design of all pieces into a single "package" - a situation which is not available today - the practical way of integrating the various separate, unrelated pieces into a workable environment is shown in Figure 2.

The objective which receives the lightest treatment in the Figure 2 approach is utilization of the Directory portion of the DD/D. Although the Directory is still present, its primary benefits depend on manual selection and interpretation of its information. It should be pointed out that although the Figure 2 environment appears to be very complex, in reality it involves the creation of only three or four modules of computer code. The DBMS shown four times is, as explained, the same module used for several different purposes. The other modules labelled as "standard" are available as routine software products, usually from the hardware vendor. The modules which will need to be provided by the using institution are the Preprocessor, the Loader, the Dynamic DBMS Interface and possibly the DD/D update program. However, there are good alternatives available in addition to doing "in house" design and coding. The DD/D update program is very similar to the update program used for a content type data base and can be written by applications programmers; or alternatively, a package DD/D can be procured which includes a data manipulation translator (language). The Preprocessor is already used in a similar fashion by some DBMS packages and can be either modified or extended to interface with the DD/D in the proper fashion and made to put out logically instead of physically translated requests. The Loader is very small and can be written in a few hours by a well qualified systems programmer. The Dynamic DBMS Interface is usually very similar to code that exists in the DBMS vendor's preprocessor, or in a static interface provided by the vendor, or both. This code can be consolidated and restructured to work dynamically, and can probably be done by the DBMS vendor in some cases. Of course, the easiest alternative is to examine the shops of other users (of the same DBMS) and buy the necessary modules to build the desired environment. This later choice is becoming more and more feasible as more institutions are discovering the value of the DD/D applied in this manner, a percentage of which chose to develop their own solution and add it to the marketplace.

Once the environment is planned, the DD/D description (attributes) must be determined. Since DD/Ds really do vary so much from institution to institution, a exhaustive list of attributes is not suggested here. However, sample data collection forms showing some of the attributes used at Indiana University are listed in Appendix B. Other sources of reference are the Guide International DD/D Project Report and the November, 1974 issue of Richard Canning's EDP Analyzer.


b.
Data Base Description Statements
Data Manipulation Language Translator
Source Code with DBMS Command
c.
Compiler and Linkage Editor
a.
DD/D Definition Statements
Data Definition Language Translator
DD/D
Object Code & DB Access Code
Data Base Management System
d.
Data Base

a. The DD/D is defined and established by use of a data definition language.

b. The data base is defined or described by use of a DD/D manipulation language or a data base description language. The DD/D receives the definition parameters for the content type data base.

c. High level source code programs with DBMS directed commands are compiled and link edited into executable code. The DD/D provides all parameters needed about the data being used. The DD/D receives information about the functions of the program regarding the data being referenced.

d. The DBMS translates the source programs commands into I/O requests or processing/I/O requests and performs requests. The DD/D provides directory information on the physical environment for last minute bridging between it and the logical requests.

FIGURE 1: The Conceptual DD/D - DBMS Environment

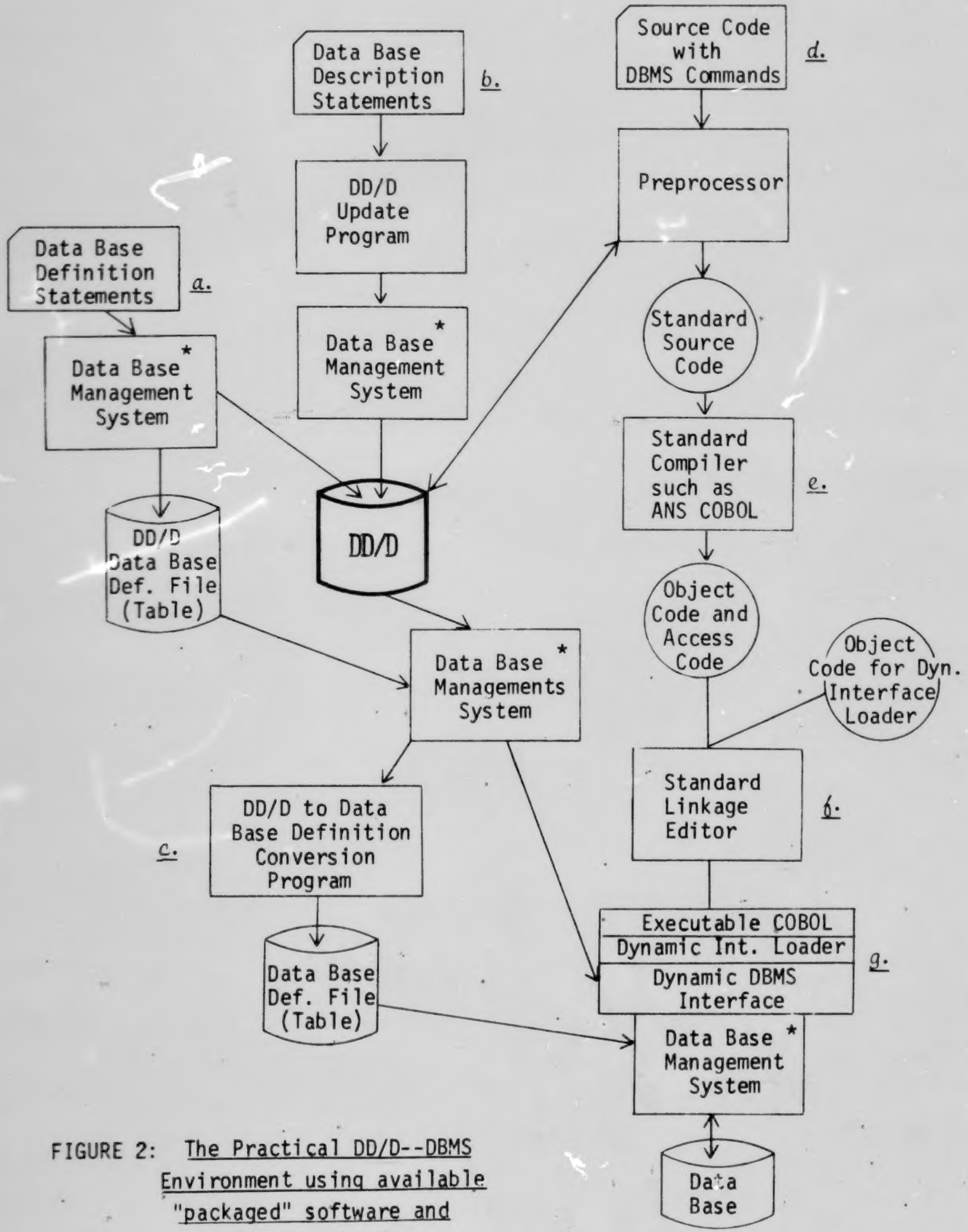

Data Base Description Statements
b.
Source Code with DBMS Commands
d.
DD/D Update Program
Preprocessor
Data Base Definition Statements
a.
Data Base Management System *
Data Base Management System *
Standard Source Code
Standard Compiler such as ANS COBOL
e.
DD/D Data Base Def. File (Table)
DD/D
Object Code and Access Code
Object Code for Dyn. Interface Loader
Data Base Managements System *
Standard Linkage Editor
f.
DD/D to Data Base Definition Conversion Program
c.
Executable COBOL
Dynamic Int. Loader
Dynamic DBMS Interface
g.
Data Base Def. File (Table)
Data Base Management System *
Data Base

FIGURE 2: The Practical DD/D--DBMS Environment using available "packaged" software and related technology

* All DBMS's shown are in reality the same software product

a. The DD/D is defined and established by use of the data base definition facility of the DBMS.

b. The data base is defined or described to the DD/D by use of an update program through the DBMS. The DD/D receives the definition parameters for the content type data base.

c. Information from the DD/D is extracted through the DBMS by a program which converts the DD/D parameters about the content type data base into the definition file that the DBMS depends on to process that data base.

d. High level source code programs with DBMS directed commands are compiled by a preprocessor which converts the DBMS commands into "standard" source code which reflects the intentions of the program to the Dynamic DBMS Interface Module. The DD/D provides all parameters needed about the data being used. The DD/D receives information about the functions of the program regarding the data being referenced.

e. The preprocessed source code is compiled by a "standard" Cobol or other type compiler into standard object code.

f. The standard object code is link edited by a "standard" linkage editor with the object code of a small loader routine to "connect" the program with Dynamic DBMS Interface Module at execution time.

g. The Loader routine connects the executable program with the latest version of the Dynamic DBMS Interface Module. The Dynamic DBMS Interface Module serves as a translater from the logical processing requests of the program to the physical processing requests need by the present version of the DBMS. The DD/D serves as a source of information to assist this translation.

The DBMS uses its "built in" data base definition file or "table" and the physically specified access requests from the Dynamic DBMS Interface Module to service the program's request from the data base.

APPENDIX A - DED Data Collection Forms

In preparing for the implementation of a university-wide Management Information system, Indiana University developed a data element dictionary (DED). The DED was implemented using four program connected ISAM files and was oriented toward describing the existing environment made up of "fields" and files, etc. As this paper suggests, the input was collected by distributing the work over the largest number of people possible and by assigning each collection task to the person who already knew enough about that particular file, etc. to quickly complete the task. The major portion of the collection efforts required only the first four forms included in this appendix. The other forms are included, however, to provide additional examples of the kinds of attributes one might want to include in the DED design.

The DED was programmed in standard languages such as ANS COBOL and did not require a DBMS. Its major value was provided directly to people who needed to know what the state of the current systems and data was as well as what existing data could be carried forwarded into the MIS. Although these benefits were considered to be significant, it is important to note that none of the software interfaces suggested in Figure 2 were utilized with this DED version of the longer range DD/D.

Although the many reports made available through this DED were very valuable in MIS planning, their inclusion here is prohibited by both the number of pages and reproduction difficulties. However, copies are available on request.

APPENDIX B - DD/D Data Collection Forms

The final DD/D implemented at Indiana University does follow the basic design and environment depicted in Figure 2. Since the DD/D in this form had to be concerned with data base and directory types of information, the number of attributes increased significantly. The three input forms which follow in this appendix are provided as a sample of those added attributes. For additional samples the Guide International DD/D Project Report is very helpful. Some additional documentation is also available from Indiana University on request.

EDITOR'S NOTE: Several sample input forms were submitted with Appendix A and B and are available through the CAUSE National Office to the interested reader.

MARK IV WITH IMS

Vinod Chachra
Systems Development
Virginia Polytechnic Institute and State University

A. Wayne Donald
Systems Development
Virginia Polytechnic Institute and State University

MARK IV WITH IMS

For several years Virginia Tech has been using IBM's IMS (Information Management System) software package as its primary data base management system. Recently, we have acquired a software package called MARK IV. The purpose of this presentation is twofold. First, we wish to discuss why it was necessary for Virginia Tech to acquire the MARK IV package, given that we already had IMS. Second, we wish to discuss some of the implementation procedures and standards followed in order to synchronize the use of MARK IV with IMS. I will be dealing with the first question and Mr. Wayne Donald will be dealing with the second one.

In order to completely appreciate the rationale for the decision for acquiring MARK IV, it is necessary to understand the environment in which Virginia Tech develops its administrative data processing system. Approximately five years ago, in order to take advantage of the advancements in computer technology, a Systems Development Department was created at Virginia Tech. This department was vested with the responsibility of designing and developing new software for administrative data processing systems. At this time, certain operating procedures were laid down. The three most important of these procedures were:

1. The Systems Development Department would not be responsible for any day-to-day operations of existing or new data processing systems.

2. It would be the administrative offices themselves, rather than some group within the Computing Center, that would actually run the data processing jobs.
3. All development would be done under IMS.

These three requirements greatly influenced both the procedure and the design of the administrative data processing systems. It also defined the organization of the department. The current organization chart for the Systems Development Department is shown in Figure 1. The primary development work in any given area is done by the project managers. Programming help for all the projects comes from the programming pool. The Data Base Administrator and the Supervisor are staff positions with the following responsibilities:

1. data administration
2. technical support
3. setting operating standards
4. reviewing data base designs
5. providing internal education

Since it was initially decided that the administrative offices would run their own data processing systems, it became necessary to keep the user office actively involved in every stage of the development of the system. This constant interaction between the project manager and the user office, though time consuming, built a certain amount of confidence in the user and insured that the system did what the user wanted it to do. In the five years that elapsed, numerous programs have been written in the following areas:

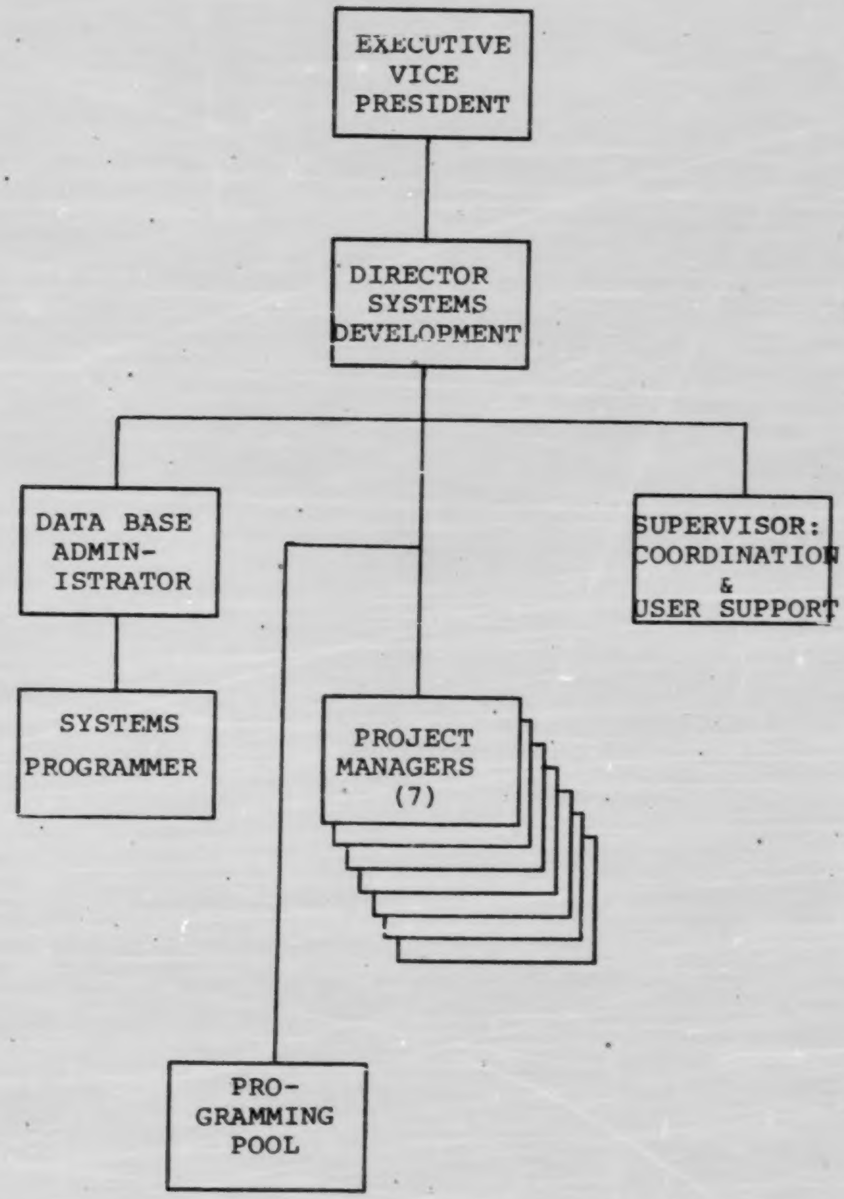

EXECUTIVE VICE PRESIDENT
DIRECTOR SYSTEMS DEVELOPMENT
DATA BASE ADMIN- ISTRATOR
SUPERVISOR: COORDINATION & USER SUPPORT
SYSTEMS PROGRAMMER
PROJECT MANAGERS (7)
PRO- GRAMMING POOL

FIGURE 1. ORGANIZATION CHART

Accounting, Personnel/Payroll, Student Records, Housing, Registration, Bookstore Management, Test Creation and Grading.

The completion of many of these systems has given rise to two rather diverse and interesting developments. First, it has created a group of administrative users whose major expertise is not in data processing but in the particular administrative area in which they are involved. To the Computing Center, the administrative users are essentially like the research and instruction users. Thus the administrative offices are encouraged to be self-dependent in their computer operations. It was found that this self-dependency could not be fully achieved without their having the capability to do certain manipulations with their own data. These manipulations could produce reports either for diagnostic purposes or for decision making purposes or even to fulfill certain ad hoc requests from special interest groups. It was found that the capability of MARK IV to generate reports from IMS data bases would adequately meet this requirement.

The second and perhaps a more readily recognized situation is that of data contamination and inconsistency. The administrative systems we have are dynamic in that data is continuously being updated in a batch and on-line mode from a variety of devices and locations. The on-line data is being shared amongst several users. From time to time the data that is being entered is either bad data or is inconsistent with data that already exists. It may be argued that a properly designed system should detect contamination and disallow inconsistencies. However, it is a fact of life that contamination and inconsistencies do occur from time to time. [One reason for this is that large systems are generally implemented in phases. Phase 1 may be implemented with a data item that needs to be consistent with something in Phase 3 and Phase 3 happens to be two years away. Finally, when Phase 3 gets implemented, nobody takes the trouble to go back and put the consistency check in Phase 1.] It became necessary, therefore, to initiate a consistency and contamination checking program for all our data bases. Once again, the data access methods of MARK IV, with certain limitations, permitted a quick and efficient method of initiating this program.

Along with these two important uses of MARK IV, we feel that the usefulness of MARK IV will increase as the data management associated with the various functional areas gets completed and the Management Information System matures to a point where it supplies data for executive decision making. (Figure 2) The data requirements for such a stage is characterized by a need for "unstructured" or demand reports. Typically, these reports will have to be produced at short notice, at high priority, and will generally require the synthesizing of several information bases to develop the required material. Such requirements would normally preclude the use of normal programming procedures. Some time in the future this will probably be the heaviest use of MARK IV.

Summarizing, then, we feel that MARK IV will serve the following purposes:

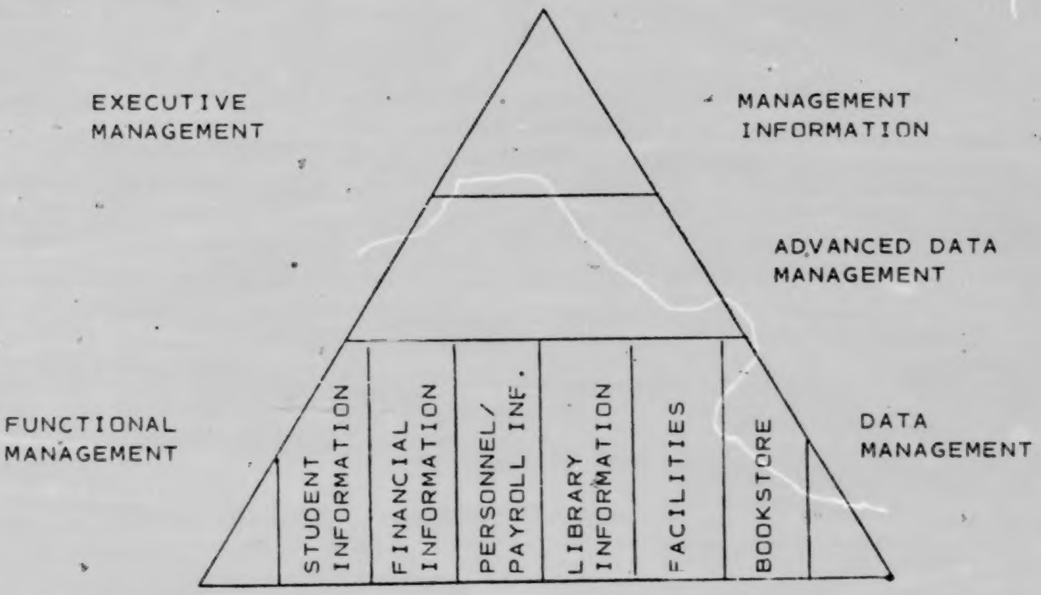


FIGURE 2. FUNCTIONAL INFORMATION AREAS

1. It will allow the administrative offices to independently manipulate their own data.
2. It will allow the data base administrator to periodically check the data base for inconsistency and contamination.
3. It will allow the generation of reports for executive decision making.

PART II

The following sections of this paper will deal with some of the implementation procedures and standards followed in order to synchronize the use of MARK IV with IMS. Specifically, the following will be discussed:

1. MARK IV Naming Conventions
2. Mappings for MARK IV Names and IMS Names
3. Implementation Considerations for IMS MARK IV Users
4. Security Considerations
5. Educational Programs
6. Concluding Remarks

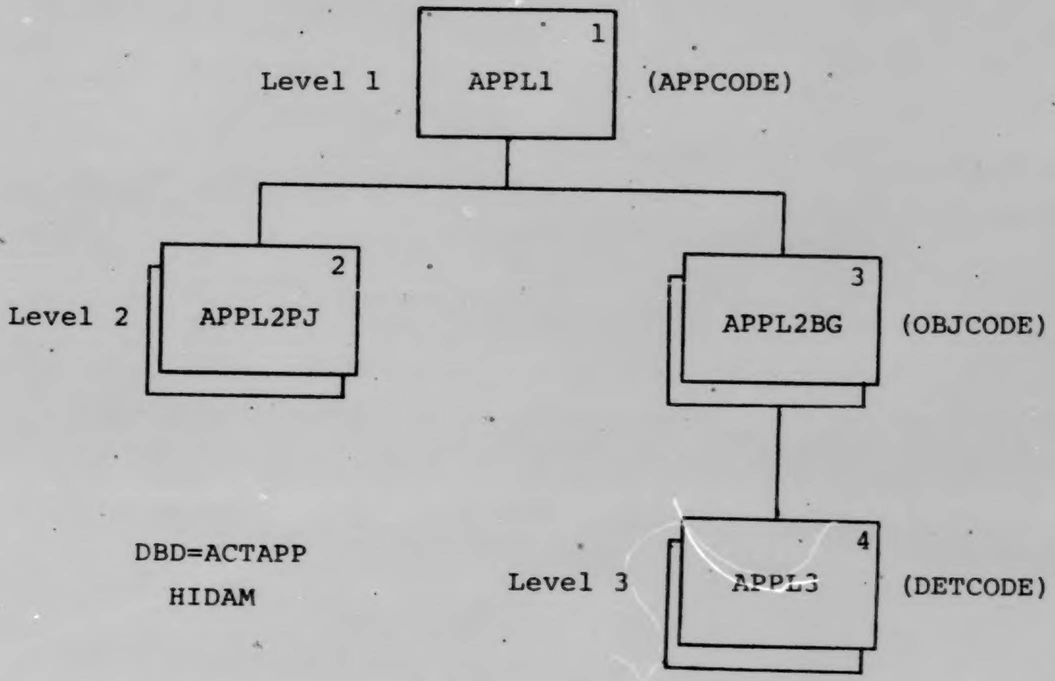
MARK IV NAMING CONVENTIONS

When IMS was implemented at Virginia Tech, a data base committee was established to review the various problems for implementing the data base management system. One area of concern was the naming conventions to be followed for data elements in the various data bases. Accordingly, the data base committee established general guidelines to be used in the naming of data elements. Since COBOL was the primary language at Virginia Tech, the data element names were designed to meet the COBOL restrictions. The guidelines were very comprehensive. Standard abbreviations were established in the use of data element names. Several specific categories of data elements were recognized and conventions established for them. Consideration was given to the first part of the names to insure that they would appear together in an alphabetic data element dictionary. These guidelines are published in an internal document (CCS-72-021) and were made available to IMS users for the definition of their data bases.

With the introduction of MARK IV as a report generator for IMS users, it was felt that similar guidelines should be established for MARK IV. A committee similar to the IMS data base committee was formed to establish these MARK IV naming conventions. The eight-character limitation of MARK IV field names was perhaps the main problem in establishing a proper set of standards. The standardization task was further complicated by the requirement that unique names be given to each field within a MARK IV file definition. Nevertheless, the committee recommended the following standards:

1. The file name for the MARK IV file definition would be the same as the DBD name of the IMS data base.
2. The IMS data base element name would be used in MARK IV whenever possible.

3. If there were a problem due to the duplication of a data element name, an attempt would be made to eliminate the duplication by using the segment number and/or the segment level number. If this were not sufficient, then an arbitrary, but meaningful, suffix could be used to establish the uniqueness of the data element name within a given file definition.
4. The comment section of the MARK IV file definition was to be used to contain the actual IMS data element name. Since this comment section was the default for column headings for MARK IV generated reports, it was decided that the user could override this on the output if a different column heading was desired in the report.
5. The PSB name to be used with the MARK IV file definition would have the format of XXXM4YYY where the X's represented a unique number for that PSB. This naming convention would identify each of the PSB's with a specific development area and would also help establish certain access security in the future.
6. Each MARK IV file definition was to be accompanied by a hierarchic layout of the corresponding data base. Each data base segment would be represented by a block on this diagram and would be numbered sequentially, beginning at number one, going from top to bottom, left to right. The level number would be indicated to the left of the block. In each block the appropriate segment name (as it appeared in the IMS DBD) was to be indicated. Finally, each key field name as specified in the IMS DBD was also specified



APPROPRIATION DATA BASE

FIGURE 3. HIERARCHIC DATA BASE LAYOUT

within the block. Figure 3 illustrates this type of hierarchic data base layout.

In order to insure that these standards are uniformly met, one individual was assigned to be the MARK IV coordinator with the responsibility and the authority to impose these guidelines whenever necessary. These standards were, however, required for MARK IV file definitions of IMS data bases. Non-IMS users were not required to conform to these standards.

MAPPINGS FOR MARK IV NAMES AND IMS NAMES

It has been indicated earlier that the comment area in the MARK IV file definition was to contain the IMS data element name. Thus, a MARK IV user would automatically have for each MARK IV data element name a corresponding IMS data element name. This provided a mapping of the MARK IV name into the IMS names. However, there was a requirement to go in the other direction. It was necessary to map the IMS names into the MARK IV names. Since the MARK IV names were only eight characters long, it was found convenient to use card columns 72-80 in the copy of the I/O area of each data base. Thus, each time a copy of the I/O area for the IMS data base was called, the MARK IV names would be automatically available in card columns 72-80. Figure 4 is an I/O area with its MARK IV names as it would appear when used in a COBOL program.

```
    COBOL NAMES                                                MARK IV NAMES

01  APP-RECORD.
    03  ACT-APP-ROOT.                                            APP-ROOT
        05  ACCOUNT-FLAG            PIC X.                       ACCTFLAG
        05  ACCOUNTING-CODE-APP.                                 APPCODE
            07  AGENCY-NUM          PIC XXX.                     AGY-NUM
            07  APPROPRIATION-NUM   PIC XX.                      APP-NUM
            07  ACTIVITY-NUM        PIC X(4).                    ACT-NUM
            07  DEPT-NUM-STATE      PIC XXX.                     DEPT-NUM
            07  PROJECT-NUM.                                     PROJ-NUM
                09  FUND-PROJECT    PIC XX.                      FUND-NUM
                09  FILLER          PIC X(5).                    FILLER1
            07  FILLER              PIC XX.                      FILLER2
        05  ALLOT-AMT               PIC S9(7)V99.                AL-AMT
        05  BALANCE-BEG-MO          PIC S9(7)V99.                BAL-MO1
        05  ENC-OUTSTANDING         PIC S9(7)V99.                ENC-OUT1
        05  EXPEND-YTD              PIC S9(7)V99.                EXP-YTD1
        05  REV-YTD                 PIC S9(7)V99.                REV-YTD1


    03  ACT-APP-PROJECT.                                         APP-PROJ
        05  PROPOSAL-NUM            PIC X(5).                    PROP-NUM
        05  GRANT-AMT               PIC S9(7)V99.                GT-AMT
        05  ALLOT-DATE.                                          AL-DATE
            07  DATE-BEGIN.                                      AL-BEGIN
                09  YR              PIC XX.                      YR21
                09  MO              PIC XX.                      MO21
                09  DA              PIC XX.                      DA21
            07  DATE-END.                                        AL-END
                09  YR              PIC XX.                      YR22
                09  MO              PIC XX.                      MO22
                09  DA              PIC XX.                      DA22
        05  EXPEND-YP               PIC S9(7)V99.                EXP-YP2
        05  REV-YP                  PIC S9(7)V99.                REV-YP2
        05  OVERHEAD-TYPE           PIC X.                       OH-TYPE
        05  OVERHEAD-PERCENT        PIC S9(3)V9999.              OH-PCENT
    03  ACT-APP-REMARKS REDEFINES   ACT-APP-PROJECT PIC X(52).   APP-RMKS
    03  ACT-APP-BUDGET.                                          APP-BG
        05  OBJECT-CODE.                                         OBJCODE
            07  OBJECT-CODE-MAJOR   PIC XX.                      OBJ-MAJ3
            07  OBJECT-CODE-MINOR   PIC XX.                      OBJ-MIN3
        05  BALANCE-BEG-MO          PIC S9(7)V99.                BAL-MO3
        05  ENC-OUTSTANDING         PIC S9(7)V99.                ENC-OUT3




    03  ACT-APP-DETAIL.                                          APP-DET
        05  ACCOUNTING-PERIOD.                                   ACCT-PER
```

FIGURE 4. COBOL I/O AREA

MARK IV IMPLEMENTATION

The following implementation considerations were developed in order to give impetus to the use of the MARK IV system.

1. Every IMS data base was required to have a corresponding MARK IV file definition. The IMS DBD name associated with that file definition would be used in MARK IV as the file name and file identification. If more than one file definition is used per DBD name, an appropriate suffix would be used on the file name to make it unique.

2. MARK IV file definitions for IMS data bases would be done within the Systems Development Department by the appropriate project managers. Glossaries of these file definitions would be made available to all administrative users to be used as desired.

3. Every MARK IV file definition, transaction definition and translate table would be put in an internal library that has several levels of account and password protection.

4. Appropriate catalog procedures were set up for MARK IV users to reduce the burden of writing the necessary JCL to run the job.

5. Appropriate options were defined to insure the proper use of data available to the MARK IV system.

6. A weekly meeting was set up to provide a forum for the exchange of ideas and problems in the use of the MARK IV system.

7. A mechanism was set up to gather various statistics on MARK IV runs. When sufficient data becomes available, it will be used to gauge the efficiency of the MARK IV system.

SECURITY CONSIDERATIONS

The committee set up certain restrictions to protect the access to the various data bases and administrative systems.

1. The IMS security already established would be applicable to the MARK IV system.

2. The detail MARK IV JCL would not be available outside the Systems Development Department. The catalog procedures set up for the IMS MARK IV users would point only to the MARK IV libraries maintained internally. Therefore, to use private libraries, the JCL would have to be modified accordingly.

3. The MARK IV program specification block would have to be approved by the data base administrator thereby permitting a check on data base activity.

4. In the future own code subroutines would be incorporated to periodically intercept the PSB name to monitor the type of activity by each user.

EDUCATIONAL PROGRAMS

The Office of Systems Development established two educational programs to be used for MARK IV. One program was for internal use, the other was for presentation to prospective administrative users. The internal program primarily used the audio-cassette course that was purchased from Informatics, Inc. After the cassette course was completed, sessions were held to discuss MARK IV usage within our environment. The University also took advantage of the systems engineer's visit and included these in the internal educational programs.

The educational programs for the administrative users were set up by the MARK IV coordinator and was intended to indicate to the users how they could use MARK IV as a report generator with IMS. There were no references to update procedures or transaction definitions. Also many of the special features in MARK IV were only mentioned and not covered in detail. The users were informed that if they had a need to use some of the other more complex features of MARK IV, they should contact their project manager for assistance. The class was held in only four two-hour sessions and a follow-up session was conducted one month after the completion of the class. This was to answer any questions concerning their use of MARK IV and to present them with some of the information from internal MARK IV meetings.

CONCLUSION

Even though our exposure to MARK IV has been rather limited, we feel confident that MARK IV can be used with IMS to fulfill a variety of needs. The combination of MARK IV and IMS synthesizes the powerful access methods and telecommunication facilities of IMS with a user-oriented report writing capability of MARK IV. This combination, we feel certain, will give us the power and flexibility to meet the changing needs for information within the University environment.

DATA BASE DEVELOPMENT IN A MULTI-CAMPUS UNIVERSITY

Richard L. Mann
Director
Management Information Systems
University of Illinois

Dorothy J. Hopkin
Assistant Director
Management Information Systems
University of Illinois

DATA BASE DEVELOPMENT IN A MULTI-CAMPUS UNIVERSITY

I. This presentation will cover the activities of one multi-campus institution and its efforts in developing common data bases, a data base administration function, university-wide applications, and an over-all management information system.

Recent research has indicated that many colleges and universities are engaging in the development of a management information system. Of 722 institutions surveyed in 1973, nearly 70%, 296 schools, reported engaging in either the operational or planning phase of an MIS project. Of these 296 schools, 88% reported that they at least were using integrated (logically or physically) data bases and common data element definitions. These two components of a management information system would seem to be the most important factors in developing a university-wide MIS. Thus it is assumed that the development of a common data base with standardized definitions is one of the key prerequisites to developing a comprehensive university-wide management information system.

The survey also found that multi-campus institutions tend to lag significantly behind single campus schools in developing a management information system. A number of causal factors may be involved:

1) The organizational complexity of multi-campus institutions. The system level MIS group attempting to develop a university-wide MIS faces a number of organizationally related problems:

   a) Many more offices (across the campuses) are required to arrive at a consensus on such things as systems design, data base design, definitions, etc.

   b) Differences in campus organizational patterns may make MIS needs, priorities, and responsibilities vary campus by campus.

c) The campus/university political climate may vary from one campus to another.

d) The nature of the over-all university campus organization. Do campus offices have any direct functional line of responsibility to the system or just to their Chancellor?

e) Academic freedom for administrative offices -- Administrative offices on many campuses feel they have the right to run their operation as they see fit. In a multi-campus institution, such attitudes are magnified when consensus is sought across many such offices on member campuses.

2) Hardware and software incompatibility. Many member institutions in multi-campus schools have developed their own administrative computing capabilities and frequently there is little compatibility between the configuration of software, hardware, and application systems which one campus has as compared to its sister campuses.

3) Geographical dispersion and campus mission. Institutions may be geographically dispersed across the state or other regional area. Such dispersion results in a number of differences in the nature of the member campuses such as the metropolitan institution versus the rural school, and the residential versus the commuter institution. The concept of out-of-sight-out-of-mind is often found in multi-campus schools. That is, if the system office is located at some geographical distance from the member campus, the campus may feel that it cannot rely on the system office and that activities must be maintained locally, and that the system office will not or can not effectuate the MIS which has been planned. Other administrative differences also contribute to the complexity of multi-campus MIS development. The most striking of these is the quarter system versus the semester system and the difference in philosophy of a liberal arts college versus a comprehensive Ph.D.-granting institution.

4) System versus campus data processing departments. Often there is no central data processing management in a multi-campus university. Data processing directors may report to either their campus management or to the system office. Coordination and cooperation in such an environment is difficult to achieve and not unfamiliar to individuals working in multi-campus systems.

5) The concept of the data base administrator is foreign to most institutions, be they multi-campus or single-campus. In a multi-campus institution, however, it becomes much more important. The ability to have commonly defined data elements, to structurally organize them into a set of common data bases, and to have common applications processing against them is a task which falls to the data base administrator in terms of control. Suddenly the data base administrator becomes a key and critically important function which was not there before and this rise in power may be resented by the various organizations with which the data base administrator must deal.

There are perhaps three general models of multi-campus MIS or data base administration organizations. The first, the Data Collection Model, centers on an institution or institutional system which may have many differentiated campuses with varied computing hardware and differentiated software applications. There may be so many campuses in the system that no other approach other than data collection from these member campuses in some commonly defined format is feasible. Thus the data is collected across the university, usually in paper or possibly machine readable form, and the university system (MIS) office maintains summary data bases which are built from the information provided by the member campuses. No control over application processing nor over the way the data formats are maintained at the local campuses is engaged in by the university MIS office. The deficiency in this model is the fact that the data may have been mapped or arrived at much differently on the various member campuses, and may not be a true reflection of commonality across all the institutions.

The second model is Common Campus Systems. Here we have commonly developed applications across all the institutions, but hardware and software systems, while maintained at each campus, are at least minimally configured to support these common applications. Because the hardware and software systems are not linked to a network, data collection and standard formatting approaches are still utilized by the system office, and university summary data bases are maintained. However, this model is more effective than the straight data collection model in that the summary data bases may be considered to be more accurate since a common application has been processed on each of the campuses creating the various data that is input to the summary data base.

The third model would be what we at the University of Illinois call the Common University Systems Model. In this model, administrative data processing hardware and software are linked across all the ADP shops and the system level MIS office to form a completely linked network. Applications, especially university-wide applications, are developed commonly and are utilized by all of the campuses. Data bases and data files are maintained at a central computer center accessible not only by the responsible campus, but also by the university MIS organization accessing information for university-level management planning and analysis purposes. This model has the advantage of having data bases designed not only to support campus operational requirements, but also to support MIS requirements. One central data base administration group can administer the data bases under a data base management system; direct access for MIS requirements is available; and each campus has access to all the software and hardware capability inherent in the central sophisticated system. The University of Illinois has taken the latter approach, and we believe it is in the process of paying considerable dividends.

II. The University of Illinois Organization

The University of Illinois is made up of three campuses--two located in Chicago (the Chicago Circle and Medical Center campuses) and a large residential campus in Urbana. The President of the University system, plus three Vice Presidents, report directly to the Board of Trustees. Each campus has a Chancellor who reports directly to the President. The President/Chancellor relationship results in a separation of authority/responsibility between the staff of the Vice President for Planning and Allocation at the system level and his staff, and the Vice Chancellors at each campus and their staffs. Only in the case of financial and business affairs is there a direct authority relationship between campus offices and a university level staff office. See Figures 1 and 2.

The ADP network is based on a three-megabyte IBM 370/158 currently located at Urbana. Four major remote locations are linked to the system: Urbana campus, with an IBM 360 Model 22, an IBM 360 Model 22 at Chicago Circle, and an IBM 360 Model 22 at the Medical Center. The MIS office has an IBM 2922. The IBM 158 will eventually be moved to Chicago. There is approximately 130 miles between the Urbana location and the two Chicago locations. Two high-speed lines support the RJE terminals and numerous low-speed lines for low-speed teleprocessing application are provided. See Figure 3.

The MIS office has three major responsibilities: the development of university-wide application systems; the development of a data base administration function and the control of that function; and the retrieval of information for analytical and reporting purposes.

For the purpose of this discussion, data base administration is defined to include the definition, organization, and design of data bases, the administration and operation of those data bases, the tuning and performance monitoring of both the data base management software and applications and the interface between users, systems analysts, and designers, programmers, systems programmers, and machine operators.

UNIVERSITY ORGANIZATION

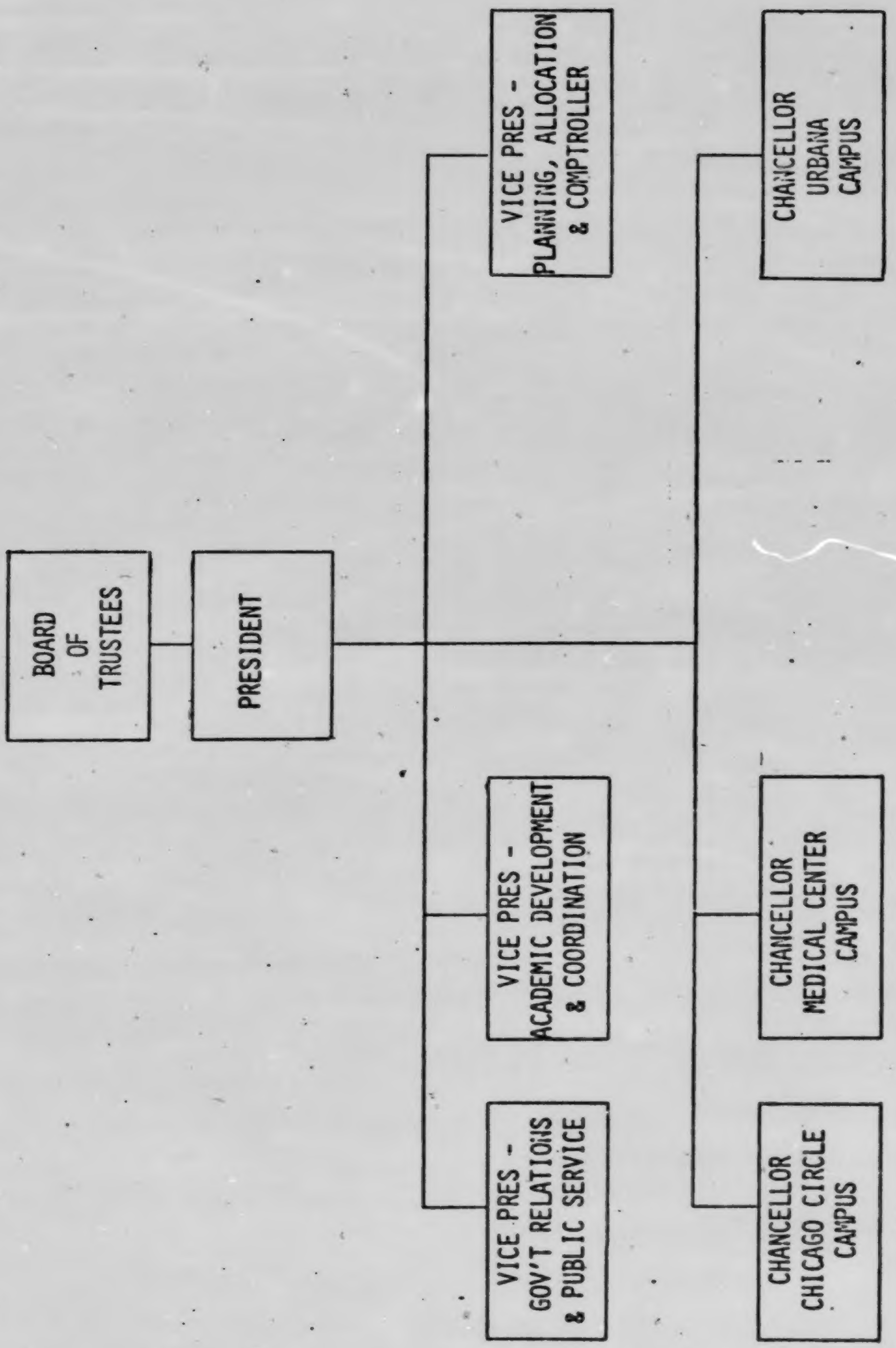

BOARD OF TRUSTEES
PRESIDENT
VICE PRES - PLANNING, ALLOCATION & COMPTROLLER
VICE PRES - ACADEMIC DEVELOPMENT & COORDINATION
VICE PRES - GOV'T RELATIONS & PUBLIC SERVICE
CHANCELLOR URBANA CAMPUS
CHANCELLOR MEDICAL CENTER CAMPUS
CHANCELLOR CHICAGO CIRCLE CAMPUS

Figure 1.

VICE PRESIDENT PLANNING AND ALLOCATION ORGANIZATION

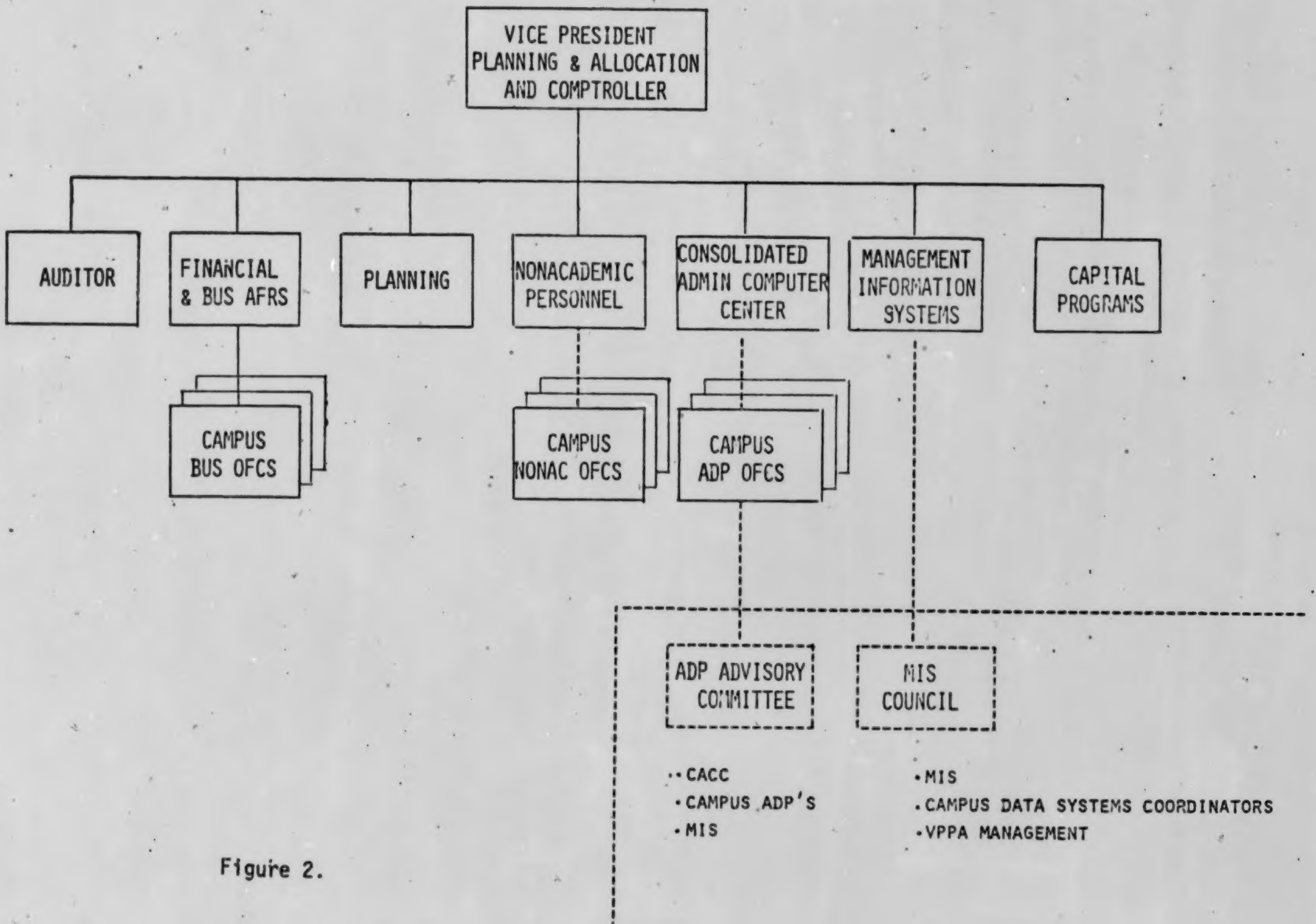


Figure 2.

UNIVERSITY OF ILLINOIS RJE NETWORK

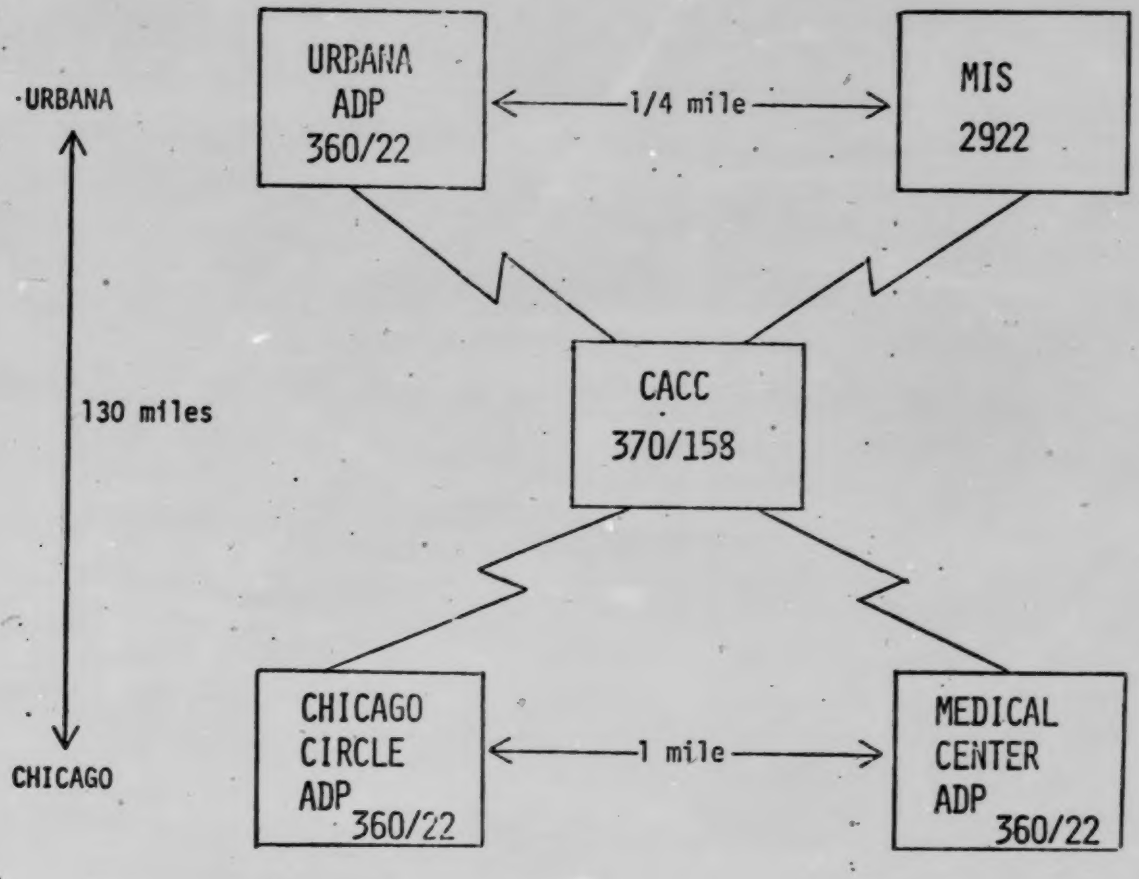


Figure 3.

III. History of MIS Data Base Development at the University of Illinois

Until August 1972, a university-level administrative data processing office to which the three campus administrative data processing offices reported was in existence. Data base commonality and application commonality was maintained by this university administrative data processing office and administrative procedures were effected to assure that applications remained as common as possible and that data elements remained as standard as possible. Because a direct line of authority between the director of the university data processing office and the campus data processing offices existed, administrative controls were reasonably sufficient to maintain this kind of commonality.

In Spring 1972, however, a complete reorganization of the University took place, including a new President and Vice President. The previous University Administrative Data Processing Director had left, and what was the Administrative Data Processing office at the university level was then divided into the Computing Center and the Management Information System (MIS) office. The MIS office was organized in August of 1972. The only direct tools for data base administration which existed at that time and which exist to this date was a data element dictionary on some 60-odd files that relied on monitoring by the university level office of campus ADP activities, and a code control book which actively monitored two common university codes--College/Department/Division Codes and Curriculum Codes. Other than these two tools, data base commonality and data base administration did not effectively exist.

In November of 1972, the RJE network was established and MIS acquired MARK IV for retrieval and analysis of data existing in our numerous files throughout the University. The acquisition of MARK IV was a holding action to allow us to get in and out of our existing data files quickly, while coming up with a long-term approach to handling our data base and MIS requirements. A top-level commitment to the data base approach by the Vice President of the University resulted in a commitment to evaluate a data base management system, to adopt a data base administration function, and to put all future University-level systems into a data base environment.

IV. Data Base Management System Evaluation

Given commitment by top management to the data base environment, the University Office of Management Information Systems (MIS) set about evaluating various Data Base Management Systems (DBMS).

The method chosen for evaluation did not entail long term benchmarks. Rather, the approach was:

1) identify what is meant by DBMS's

2) review available documentation and literature

3) discuss systems with present users

The systems chosen for review consisted of those most visible in computing environments and compatible with our IBM hardware (370/158). These systems included:

1) Software AG's ADABAS

2) IBM's IMS (DL/1)

3) MRI's SYSTEM 2000

4) CINCOM's TOTAL

The criteria for evaluation was based on features identified in the literature as well as experience gained with shortcomings of traditional data management. Those features were broken into two parts: 1) fundamental features and 2) operational features.

The fundamental features include those that address problem areas that DBMS's purport to alleviate. The operational features include those items that have a direct impact on the University of Illinois computing community. Briefly, they are:

1) Fundamental features
    - Data independence
    - Security

1) Fundamental features (continued)
    - Recovery
    - Reorganization
    - Integrity

2) Operational features
    - Vendor implications/cost/support
    - Data structures
    - Creation complexity
    - Resource usage
    - Application programming complexity
    - Systems aspects

A full discussion of these items is presented in the evaluation paper itself.

V. Data Base Management Selection and Implications

The early results of the evaluation were presented in December 1973. Work continued, looking at more details, until in May 1974 the recommendation was made and accepted to use IMS/VS data base/data communication system.

Briefly, some of the reasons for this choice were the emergence of IMS/VS as a follow-on to IMS/Version 2. Its additional features of 1) secondary indexing, 2) program isolation, and 3) variable length segments enhanced the product. Additionally, one early constraint--that of using CICS as the teleprocessing software--was lifted, making the full IMS possible.

Having selected IMS, certain ramifications presented themselves. Consideration of additional supporting software had to be made, particularly a dictionary system. The MARK IV DL/1 interface had to be purchased to provide continuing support to our users who utilize MARK IV for one-time reports and ad hoc requests.

Tools, in the form of productivity aids, were identified for ease in using the IMS system. In particular, these productivity aids include IMSMAP, DBPROTOTYPE, BTS, and 3270 format feature. Additionally, we selected IBM's IMS Dictionary System to support our IMS work. This decision to acquire the IMS Dictionary was made mainly from a standpoint of the PL/1 interface, certain on-line capabilities, and the price.

Training was identified as a major issue and certain training guidelines were established, particularly for the MIS staff. Since we had selected IMS/VS, we had some time to work with waiting for VS to be installed on our system. Following are the courses we have had and expect to have for IMS:

1) IMS For Analysts was taught by Performance Development Corporation in August, 1974. We felt that this would serve as a foundation for our people to work with IMS and to start them thinking in terms of IMS as they designed new systems.

2) In September, 1974, we sponsored IMS For Managers, taught by McDonnell/Douglas Automation (MCAUTO). Some 45 managers from across the campuses and the General University attended. The presentation included an overview of IMS and data base concepts in general.

3) Two MIS staff members attended IBM's IMS systems programming schools. These consisted of IMS Data Base Design and Implementation I and II and IMS Data Communications Design and Implementation.

4) IMS Message Format Services from IBM was attended by one staff member with the intent of carrying back the course material to the rest of the staff.

5) In December 1974, MCAUTO taught a 2-week DL/1 programming class for programmers/analysts.

In total, about $16,000 will be spent training 12 MIS programmer/analysts in IMS concepts, analysis, and programming. In the future, training will be extended, perhaps by using Control Data Corporation's multi-media courses, both for initial and refresher uses. In-house training for 3270 Message Format Services will be used since the detail course will be unnecessary for all people (since the Productivity Aid, 3270 Screen Image Processing has been acquired).

IMS/VS was installed at the Computer Center on November 18, 1974, batch only version. Our experience has, to date, been very good. The generation of the batch only system went well and our people were able to use it for problem programs during the class.

Given this foundation, the additional software, the Dictionary System, MARK IV DL/1 interface, training that we have and intend to carry out, and top-level commitment, we have set the stage for the data base administration function.

VI. The Data Base Administration Function

The data base administration function is one that is technical, administrative, and diplomatic. The organization of this function reflects these characteristics. At this point in time, our concern (and organization) is with the technical aspects of IMS and the data base environment. Characteristics of the organization will change over time and with maturity.

The technical function is one that encompasses the design and definition aspects of data base administration. This includes seeing to it that the software works when installed. It also includes the implementation of the utilities and productivity aids, establishing the rules for the use of the dictionary, and getting the initial set of data elements into the dictionary.

In addition, naming conventions are being established based on the technical aspects of IMS that we see today, e.g., DBD's, PSB's, segments, terminals, transactions, formats. We have taken these conventions to but not through the field level. We will, however, require that all fields in each segment be defined to us such that we can establish them in the dictionary and more easily create MARK IV file definitions, even though not required by IMS.

We will be reviewing the teleprocessing considerations, the technical security provisions, the logical and physical data base design and implementation rules. When the technical limits are known and the guidelines are finished, we will move more to the administrative or operations side of data base administration.

Each of the technical aspects of IMS as they are shown in the attachment have their administrative counter-part. Therefore, security mechanisms are established through IMS; security policy must be established. Our present data security statement dealing with the user environment must be enhanced to cover the data base area. We will promote data sharing by monitoring and knowing the data that is controlled by IMS. Code control will extend beyond today's college, department, and curriculum, and incorporate certain federal standards and institute control on universally used codes throughout administrative systems (e.g., sex, marital status). Administration of the dictionary will include preparation and distribution of reports from the dictionary and constantly monitoring of data being entered and defined for consistency, adherence to standards, and to promote non-redundancy.

The integrity system will be carefully watched, particularly ensuring that recovery is properly handled. We will be preparing standards for who should use the system, how to use it, and what to do. This will include the areas of application programming, application design, and operation. Consulting will be provided to all of these groups as well as to some users.

Monitoring the usage of the system for degradation or for recommending changes will be on-going. Evaluation of performance and efficiency as well as simulation will play a more important role in later stages of data base administration.

During the discussion of functions, the many interfaces of data base administration emerged. These interfaces are best shown in Figure 4.

DATA BASE ADMINISTRATION
THE VARIOUS INTERFACES

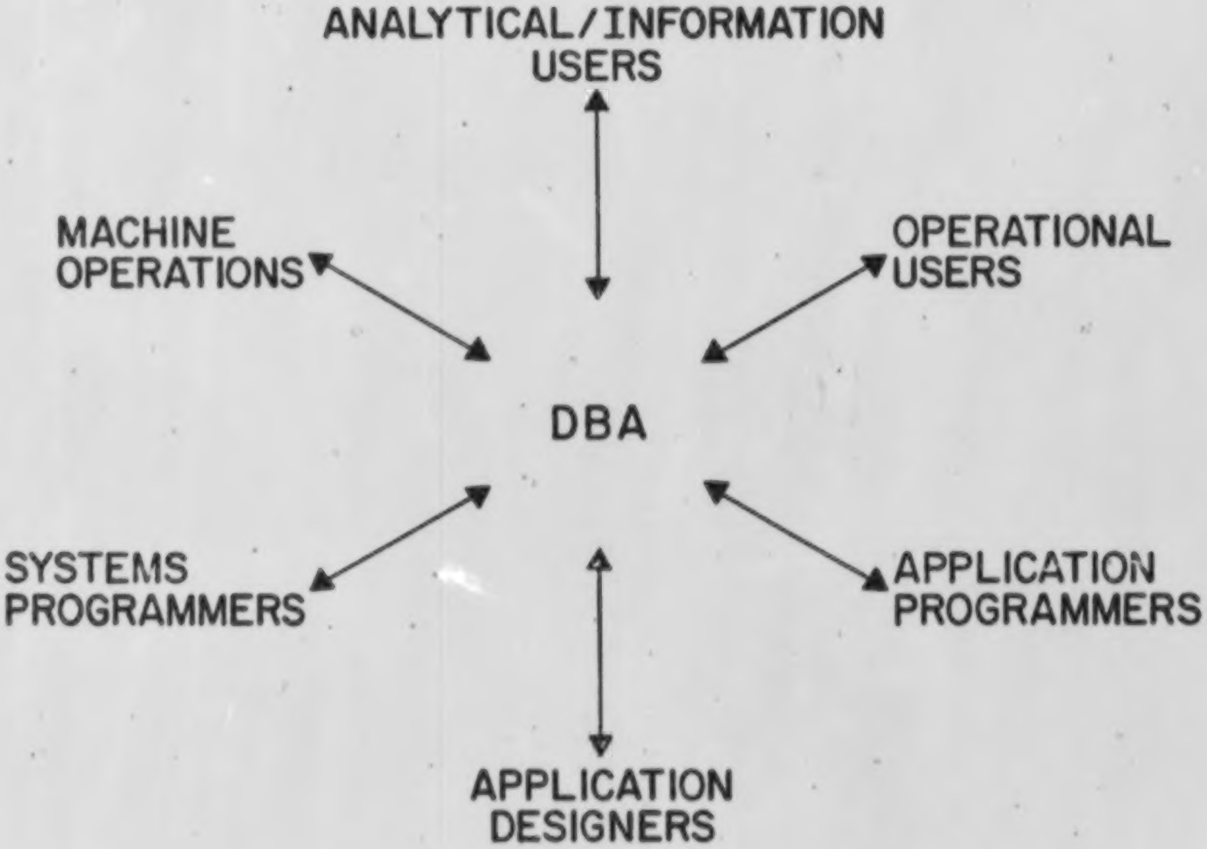


Figure 4.

VII. Interfaces That Relate To The Data Base Administration

The data base administration must cooperate and communicate with the systems programmers to ensure the software is working, to provide for the resources, to estimate resources, and to install additional software.

Guidelines that have been developed for application programmers and designers will provide the basis for consulting with these groups, both for data base and teleprocessing. The machine operation interface happens later in development phases, mainly in terms of master terminal operations where data bases are being abled and disabled, monitoring for errors, and working with the operations group for recovery from hardware failure.

Perhaps the most difficult kind of interface is with the users. They are of two kinds, those who can be classified analytical/informational users, and those who can be termed operational users. For the operational users, an educating process is needed to explain to them how their system will work in the on-line environment and how they can cause performance to degrade and asking for help from them in terms of transaction design. With the analytical users, the data that appears in the operational data bases is used to prepare studies, cost estimates and external reports for the General University. These users will need a tool for getting at the data base whether it be the on-line data base or a snapshot of the operational data.

Given the functions and the interfaces for data base administration leads us to new policies in the area of data processing as well as in the whole community.

VIII. Policy Considerations In The Data Base Administration Area

Given the preceding functions that have been mentioned and the Vice President's statement of responsibility for data base administration, it becomes clear that the environment has changed to some degree. There are many inter-related facets. It is a new environment and will create an organizational impact. We will have less autonomy on the part of developers of new systems but we believe we will be providing more capability. In order to manage the data resource, this control, responsibility, and monitoring is necessary.

The problem areas or where new policies will be needed are in the following:

1) Data definition or the standardization of data definition and codes; who defines them, arbitrating differences between them.

2) Policy as to who owns the data in the user community. In the past, files were owned by individual user offices. There was a proliferation of the data in those files as many users needed to have the use of certain portions of the data. In the future, since the data will be shared, it will be a jurisdictional question. We thus must revise our present data security statement and make the policy for the data base environment.

3) There will need to be policies as to who develops systems under the data base management system. This applies to the types of applications that are developed as well as whether local campus development can be carried out. Local campus development is expected but only under data base administrator-established rules for element definition, design review and data base administrator installed structures.

4) Additionally, in the development area, the policy must be palitable to the computer resource. The machine operations and systems programming staff must be able to maintain the volume incurred by any new systems installed under the data base management system. Also in this area we would hope that the design for the use of this system is carried out not merely as a conversion mechanism but as a redesign in order to use the tools provided by IMS.

5) Another policy that must be considered and established pertains to what tool we will use for the research and statistical gathering users that operate on operational data. The tools that must be considered are other software systems as IRS, IQF, and GIS or merely spinning off files for reporting purposes using MARK IV. First we must solve the operational data problem and then deal with the policy for using the operational data. This may well be done with frozen sets of data taken at particular periods in the year that the research people can use.

6) The growth pattern and development of systems operational under IMS must also be considered. We expect to see campus data base groups formed to use the established tools and rules as provided by the MIS office.

Thus the data base administration function has been formed and centralized. Although the group now consists of 3-1/2 people, it has been more or less divided into three major areas: those areas known as technical specialists, data dictionary/data element and code specialists, and consultants to the data processing community. We would expect that the group would grow in each area to 4 to 6 people to support data base development and activity.

IX. Application Development

One of the continuing problems of application development in a multi-campus institution is getting people to agree. For example, at the University of Illinois, a Personnel System which is currently in a design phase, requires that members of the design team must deal with eight separate offices over three campuses that are directly responsible for personnel. These are not the interfacing offices such as Payroll and Budget Offices. Adding those in, the number is closer to 15. Thus, application design and development tends to take longer in multi-campus schools because of the coordinating and consensus-types of problems which are faced.

The University of Illinois has begun to use IBM's Business Systems Planning (BSP) approach to developing a long-range plan for information systems design and development at the University. A three-week BSP customization phase was undertaken in the Spring of 1974, during which it was determined that financial systems should receive the highest priority for developing a long-range implementation plan. Subsequently, a 6-month BSP Phase II project, which ended in December of 1974, specified the first of the financial systems which should be developed, indicated a general design and approach for that system (in this case, a procurement and disbursement system), and defined generally the follow-on systems which should be built to create a true financial data base. The BSP project was extremely successful and well accepted by university and campus administrators. It is likely that a similar BSP project will be organized for a new university hospital which will be established in Chicago.

The general priorities for the University MIS in terms of application development are financial, followed by personnel, student, and then facilities. One of the difficulties in a multi-campus institution is establishing specific priorities. It appears now that the development of the procurement/disbursement system and a concurrent activity in the area of personnel seem to hold the highest priorities and will see the most activity.

X. Conclusion

Over all, there are a number of expected benefits which the University of Illinois' embarkation into the data base-MIS approach is likely to yield:

1) Uniform, consistent, accurate and timely data to provide information to internal and external agencies, to all campuses, and to the general university system.

2) Protection mechanisms to safeguard our data.

3) Responsiveness to necessary changes and additions to our data bases.

4) Promotion of data sharing between campus offices and campus and university offices.

5) The creation of a data resource that will be the foundation for true management information systems.

6) Better control at all levels from the administrator's viewpoint.

The University expects to face considerable problems in the years ahead in developing the data base approach to a management information system, but we feel it is the right direction and hope to make substantial progress within the next year.

AN EVOLUTION IN ADMINISTRATIVE DATA PROCESSING

Joanne M. Gallitano
Director
Planning and Information Services
Hunter College

DATA BASES: AN EVOLUTION IN ADMINISTRATIVE DATA PROCESSING

Before we begin you need to know that I am an evolutionist. That is important for you to know because this paper is going to present a look at both the history and the future of administrative data processing in terms of evolution. That means that the trends of history and the logical extension of those trends into the future will be explored. One key area of attention will be the role of data bases in the evolution. The primary purpose of this paper, however, is not to convience you that there has been or is an evolution going on. The purpose also is not to encourage you to analyze your computing environment for the purposes of locating it precisely in the evolutionary phases about to be described. Rather, the purpose is to use evolution as a concept to form a conceptual framework that will help us to observe our computing environment with some objective distance, to analyze it in broad terms and, as a result, to understand it, to manage it, and even to plan for it more effectively. That is, this paper presents a perspective for viewing the interrelationship among the seeming diverse daily problems we face in computing in the educational environment. If you wish, it is also a foundation for planning the evolution of computing in your environment. Beware that this type of conceptual work does not replace doing a thorough requirements study or defining clear-cut goals and objectives. It only provides some perspective for doing those all important steps.

One further word of caution before we begin. While much of the paper will define various phases along a continuum, one should be aware that reality is not composed of neat boxes in a matrix or of precise dots on a time line. A conceptual model is necessarily artificial. Such a model also does not always consist of very precise measurements of things. In the case of computing this is very much the case since computing as a discipline or industry lacks techniques for precisely measuring its many aspects.

I am going to begin by reading a quotation to you;

> Effective utilization of modern information processing methods has become increasingly important in the management of today's growing and changing universities. While these institutions have often pioneered in the use of computers for scientific applications, administrators generally have used computing facilities mainly in operational systems such as payroll and registration to process large quantities of data and repetitive transactions. A benefit also to be realized, however is that accurate, timely, and concise information can be provided to assist management in administrative decision-making and control.(1)

While preparing this paper I happened to see the June, 1974, issue of UNI-COLL Newsletter. The quotation I just read to you is from that Newsletter. It caught my attention because it seems to me to be characteristic of our current thinking about administrative data processing. There is a key thought in the middle of that paragraph:

> ...administrators generally have used computing facilities mainly in operational systems such as payroll and registration to process large quantities of data and repetitive transactions. A benefit also to be realized, however, is that accurate, timely, and concise information can be provided to assist management in administrative decision-making and control. (emphasis added)

Instinctively, we think of operational systems as somehow different from management information systems, generalized information systems or data base management systems. Somehow we know we have made progress from card oriented systems to data base systems, but what does this mean? What is the character and dimension of that progress? How do data bases relate to past and future trends? The purpose of this paper is to try to provide a perspective for more clearly viewing these ideas. Our discussion will center on three areas:

1) Development of a conceptual framework to describe the ADP evolution toward Data Base technology.

2) An example of the framework applied to a real-life case, and

3) Some conclusions.

Before describing the framework, let us take note of some difficulties we will encounter and state some limitations on the scope of our discussion. The first problem is a lack of clearly defined terms to describe various computing phenomena. For example, we may not all mean the same thing when we use terms like "operational systems" or "management information systems". I will try not to add too much to that confusion. Consequently, the term "administrative data processing" is used here to mean business data processing as it is performed in an academic institution. No attempt will be made to generalize beyond this to the totality of business data processing as it is practiced in other sections of industry/commerce.

Secondly, this paper will attempt to use current literature in the field as the basis for forming the conceptual framework of the administrative data processing evolution. This literature is, of course, scarce. Our purpose is to use the work that has been done, extend it slightly and apply it to a real-life case. From this exercise we hope to learn more about the evolution in which we are involved with a view toward improving our ability to work with it, to manage it and, perhaps, even to plan for it.

Thirdly, while the concluding section of this paper will suggest some steps to take, the primary purpose of the paper will not be to explore optimum paths through the evolutionary process or to analyze which phases of the evolution can be by-passed. Rather we will focus on describing the key characteristics in the evolution as it has occurred and is occurring in order to understand the relevancy of data bases in the ongoing process and to gain perspective for viewing our own computing environments.

A PERSPECTIVE

Recalling our quotation from UNI-COLL, we noted an implied trend of progress from operational level systems to management planning and control systems.

The concept of trend or an evolution is further reinforced by this quotation from the SILT Report on Computing in the 1980-85 Timeframe that SHARE, Inc. will release for publication this January:

> Data processing increasingly contributes directly to decision making processes, a characteristic that forces evolution in the direction of "real-time" data capture and direct interaction between the applications and their end users.(2)

To more specifically describe this evolution let us use the work of Nolan and Gibson as published in the Harvard Business Review (3). Their work is based on a survey of a number of corporations in various industries examining the characteristcs of their EDP operations. They found a number of significant similarities among the enterprises they examined. They described their findings as trends shaped as "S" curves. Since the focus of this paper is the college and university environment, I have reviewed the Nolan and Gibson framework from this more narrow point of view. Their framework with a few modifications or extensions appears to be workable for the educational environment. Some additional extensions have been made to look into the future as we see it now, which is about one year after their article was published.

In Nolan and Gibson's analysis the administrative data processing evolution is composed of four stages of growth in three areas of growth. Expressed as a matrix, one would have the stages on the horizontal axis and the growth areas on the vertical axis.

Illustration 1: ADP Evolution (Slide #4)

On the vertical axis, the growth areas are what I will call the "Characteristics of the Evolution". I have added three to the three that Nolan and Gibson describe for a total of six. Each of these characteristics will be viewed as a continuum that proceeds through time as identified by the various stages on the horizontal axis. Each stage is described by a list of key events or conditions. Really these are patterns. It should be noted that, while we have drawn discrete compartments for purposes of constructing a conceptual view, in reality the boundary between one stage and another is usually quite indistinct; moreover movement from one stage to another is rarely in a straight direction forward. In fact, one may frequently find a residue of former stages lingering behind in one area while the remainder of the enterprise has moved into another stage. Again our purpose here is to gain perspective and to form a foundation for thinking about the morass of daily problems in some orderly way. Now let us look at each continuum.

APPLICATIONS

The general trend in applications has been to move from supporting limited or discrete operational functions toward supporting increasingly comprehensive management functions. Accordingly the complexity of data manipulation operations has progressed from simple totaling to complex analyses and comparisons. We now speak of program algorythms. We will, in the future, speak of standard logic design constructs, interchangeable program logic blocks and a host of other yet-to-be conceived notions. A more detailed view is given in Illustration 2 which we might find helpful to review at this point.

Illustration 2: Applications Continuum (Slide #5)

Note that a fifth stage has been added to Nolan and Gibson's origional four stages. A number of key conditions characteristic of each stage have also been added. The trend shown here is generally described as moving from operational level systems to information processing to management and control systems and finally to total enterprise systems. The total enterprise system is a fairly new idea. It implies the treatment of all data in the context of its use by various user components of the enterprise. The enterprise may be a corporation, a college, a consortium, a conglomerate, etc. In other words, the enterprise is the universe or the population being served. A total enterprise information system may include some simulation models and may itself be developed by using such models.

The pattern of applications as shown here probably seems familiar to us all. We started with payroll or accounting or maybe student registration. The number and type of applications computerized proliferated with attendant growth in the cost and size of the computing function. Eventually the need to control the sprawling computing function became paramount. In Stage 3, this is represented as a moratorium on new applications though predominantly I think this turns out to be a reduction in the rate of new application starts rather than a complete halt. The thrust forward caused by new hardware/software and the continually rising demand for more, better and more timely information keeps moving the application development activity on into the data base stage and beyond it.

PERSONNEL SPECIALIZATION

Illustration 3 shows the details of the Personnel Specialization characteristic of the administrative data processing evolution. The trend from Stage 1 through Stage 5 is characterized by a growing number of job titles that are increasingly specialized and functionalized. For a look into the future, the fifth stage has been added to show some new specialists, such as the Data Base Administrator. In addition, some functions traditionally performed by technical personnel will eventually be performed, at least in part, by the end user, a person who is not especially trained in computer techniques or technology. (Slide #6)

MANAGEMENT TECHNIQUES

The Management Techniques Continuum is viewed as consisting of three sub-continuua: organization, control, and planning. The trend in management techniques is toward integrating EDP management and control into increasingly higher levels of the organizational hierarchy. Concurrently, the computing function has moved from a position as subordinate to the department of first application to a position as a separate entity in the organization. The budgetary control, productivity control and planning have increased in formality and detail as the EDP function progressed. A review of Illustration 4 A, B, and C will give the reader the details of these trends.

Illustration 4: Management Techniques Continuum
(Slides 7, 8, and 9)

As for the future, the trend is likely to be characterized by increasing awareness of EDP as a vital service and of the necessity to involve the non-EDP managers in the computing decisions and planning so that effective computing service can be achieved. As management's view of computing becomes more service oriented, its view of data is likely to become more resource oriented. Data will be recognized as a valuable resource to the enterprise and will be treated as such. A natural consequence of this trend is an increasing need for every manager to have some computing knowledge even though his responsbility is not for the EDP function per se.

With respect to EDP management itself the major puzzlement is finding the appropriate degree of centralized vs. decentralized control as well as determing the line vs. staff positioning of the EDP management function.

DATA MANAGEMENT

The Data Management Continuum may be viewed as consisting of these components:

1) data organization

2) access methods, and

3) data aggregation.

The trend has been toward more complex forms of data organization going from record and files to data bases and data banks. With this, data structures have moved from simple, ordered strings of data elements to complex hierarchies and networks of data elements. Also, the trend has been progressing from sequential access methods to various random, relative, direct and other types of more complex access methods. Data aggregation has moved from the minimum levels of records pertaining to single end uses to comprehensive data bases pertaining to whole enterprises of uses. This is leading to data sharing and to making increasingly larger aggregates of data available to a single application. Throughout, the data element has remained the smallest external unit of measure for data organization and aggregation, but not for access methods.

Illustration 5: Data Management Continuum
(Slide #10)

Of all the continuua we have described here, perhaps the most critical changes are occurring in the Data Management area. As shown in Illustration 5 we are progressing from simple records as the most comprehensive unit of organization to data bases and beyond. In the course of this evolution our focus is shifting from the program as the center of the computing universe to that of data as the center. The implications of this shift can be expected to ripple through all the other continuua causing changes in them. In fact, as time passes and our view of what transpired becomes more clear, we may conclude that this shift is more revolutionary than evolutionary. Charles Bachman in his ACM Turing Lecture said that this change is analgous to the Copernican theory in its impact as well as its concept.(4) We will return to this point later.

HARDWARE/SOFTWARE

The Hardware/Software trend is characterized by progressively higher speed processing, increasingly complex operating systems and system interfaces as well as by ever larger, cheaper memories. The last is probably the most important in terms of impact on our view of uses for the computer.

Software in the form of operating systems has progressed to performing increasingly comprehensive support functions for the vast array of computer applications. This includes job scheduling, data management, storage or memory management and I/O management. This area is also moving toward the increased use of firmware. Integrated uses of hardware, firmware and software functionalized to perform the various duties required to go from input to output is a key part of the trend in the later and future stages.

While a detailed drawing of this continuum showing the various machine models and listing the increased capabilities of each has not been included, the trends described here are clearly observable. The key point about systems hardware/software is the push-pull effect it has on advances in the other continuua as well as on our perceptions of the roles computers play in our daily living.

USER DEMAND/PERCEPTION

This continuum is intended to provide a description of the user of computing services to the extent that the changes that have occurred in the type of user and in the demand the user makes for computing service can be characterized. By referring to Illustration 6 (slide #11) we can see the trend has been for single user (which may be a departmental function such as accounting, purchasing, personnel) to multiply into numbers of single, unrelated users. As the use of computing proliferated, these functionally discrete users came into increasing contact with each other leading to growing interrelationships among them. As a consequence, user demand changed from transaction-oriented within the narrow scope of each user's own functional area to transaction-oriented within interrelated functional spheres.

As higher and higher volumes of data and increasing numbers of functional areas became computerized, the users' demand for information in various forms increased in both scope and complexity. At the same time the user became both more knowledgeable about the computer and more dependent on it to do his work.

There are other factors (e.g., the "keeping-up-with-the-Joneses" phenomenon, the computer as magical and infallible machine, etc.) which contribute to the demand increase, but which are not described here. For our purposes, it is sufficient to say that the future is an increasing demand for more comprehensive information, for faster access to this information, and for flexible reporting to make the information more readable for any audience of users. To put it concisely, the trend is toward decentralized, rapid access on a demand basis. (4) This trend is diagrammed in terms of the type of output produced in Illustration 7.

Illustration #7: The Computerized Information Output Onion (Slide #12)

More and more the success of computing will depend on our ability to provide the service required by the user in a manner that he percieves to be satisfactory.

SUMMARY

Nolan and Gibson described several "S" curves of growth in Application, Personnel Specialization and Management Techniques. They also stated that there probably would be more "S" curves;however, much of the discussion here has tended to flatten the peaks and valleys of the "S" curves. No precise qualifiable evidence is available to justfy either view.

The Nolan and Gibson framework was modified in this paper by including more characteristics in each trend. It was also extended by including the additional trends of Data Management, Hardware/Software, and User Demand/Perception. To summarize all these ideas let us review the following diagram:

Illustration 8: Onion & Inverted Onion (Slide 13)

The Application trend is described in this diagram as a series of concentric circles which are read from the center toward the periphery. Related to this onion are two smaller sets of concentric circles which express the Data Management Continuum. The inversion that occurs when we move from programs as center of the concentric circles to data as the center of the concentric circles is shown by broken lines. The relationships among points in the application continuum and corresponding points in the Data Management Continuum are shown by the solid lines.

Perhaps the key point is for us to know when we have made the change from "program is central" to "data is central". In the diagram the line from the Management and Control application systems circle goes to either of the Data Management Onions to show that this is usually the intersection where the need to change becomes apparent. The degree to which data really becomes central has a great deal to do with what we choose to use for the components of the Data Management and Operating Systems Interface layer of the inverted data management onion. The more capable those system components are of doing all the data management job, the more central the data will be in terms of its shareability and non-redundancy. To a very real extent the technology we presently call data base is the key to reorienting our computing capability to meet the user demand for better, more meaningful and timely information service.

What we have done so far is to view each key characteristic in the evolution of administrative computing. We have discussed the profile of each continuum viewed horizontally through time. We have noted that the advent of data base technology appears to be a major turning point in this evolution.

Now let us turn our attention to the application of this evolutionary framework to a real-life example. To do this the various continuua will be viewed both vertically and horizontally, that is, we will not only look for progress along one continuum, we will also check our location on various continuua vertically in the same timeframe or stage.

AN EXAMPLE - EIS

Educational Information Services, Inc., is a computer network formed to provide large scale computing resources and related services to the educational institutions of New Jersey.

One of the services that EIS provides is administrative data processing application systems. EIS began providing this service at a time when the leading edge of the data processing world was moving into stage 4 or 5 across all continuua as described earlier. /‾Refer to Slide 4 - Illustration 1: Overview of ADP Evolution_7 A survey of the environment EIS was to serve revealed that most of the computerized applications that existed were very single-program oriented with the exception of a few serially executable, file-oriented systems. There was a very high degree of data redundancy aggravated by the lack of consistency in data element definition and absence of interfaces among various systems using the same data elements. In other words, the applications were hovering somewhere between stages 1 and 2 while the moratorium on new applications that generally characterized stage 3 had been declared. To put it another way, the nature of computing at each institution in the environment was either in stage 1 or stage 2, but the totality of computing at those institutions as symbolized by EIS was in stage 3.

Implicit in the formation of EIS was the directive to consolidate all applications across all institutions and reduce the rapid proliferation of application that had been taking place. EIS was to import systems or adapt local systems for more general use. It was not expected to do any significant new application development. An examination of stage 3 characteristics across all continuua applied to the EIS case provides a clear profile of the situation. As compared with the summarization in Illustration 1 (Slide #4) we have:

1. A moratorium on new applications development.
2. Personnel specialization being increased for maintenance and systems.
3. Management techniques which view EDP as a separate function and which aim toward tight control.
4. In the data management area there is significant data redundancy.
5. Introduction of large scale /360 and /370 machines.
6. Separate users brought into interrelationship via EIS giving rise to increased expectations for use.

The difficulty computing faces during stage 3 is finding direction and momentum to move forward into the future. In the EIS case, it was faced with the need to consolidate stage 1 and stage 2 in order to bring the whole environment into stage 3. At the same time because of this hind sight view, EIS was faced with the nagging feeling that it had also to find a way to move the whole environment forward.

In a Five Year Plan published by EIS in February, 1973, the strategy for consolidating and moving computing forward was stated. As it related to admininistrative data processing the plan stated a key objective as fellows:

OBJECTIVE: Improve and expand EIS's Administrative Data Processing capability in the five service areas (Analysis & Design, Implementation, Installation & Support, Education, Planning).

a) During the first two years, stabilize and refine the systems and services presently offered by EIS. Develop additional technical capability to produce a broader range of services and more advanced computing systems in the future.

b) During the next three years, use this enhanced capability to produce a new array of services and systems that are a refinement of the previous years' work, that take advantage of new technology and that are responsive to new user needs.(5)

In outlining the requirements for improved service, the application area was viewed as key. The plan described a course of action that would 1) provide computing support for all of the fundamental administrative functions, 2) interface these application systems and consolidate the data elements used in each into a generalized data dictionary that could be used for general reporting and information query, and 3) build a data based application environment by utilizing the data dictionary and a data base management system. The applications built during the first phases were to be interfaced with the data base management system or replaced by new functions constructed by using the data manipulation language associated with whatever DBMS was chosen.

After two years of the Five Year Plan, EIS had implemented application systems to support the basic administrative applications and steps were being taken to begin interfacing these applications and their respective data elements.

The pattern of EIS progress, as shown in the diagram (Illustration #9: EIS Five Year Plan) was very similar to the continuua patterns described earlier in this paper. If we consider EIS's inception as the stage 1 point for itself, as opposed to EIS symbolizing stage 3 for the total environment, then an approximation of the "S" curve described by Nolan and Gibson can be observed. We can also see a leveling out of the "S" curve as the environment moves more toward data base.

Another way to use our conceptual framework is to make a vertical cut through all continuua at any point. This provides a useful checklist. For example, such an exercise for EIS might produce the finding that Stage 4 Management Techniques are being applied in an environment where the applications characteristics are equivalent to stage 2 which leads to a need to more closely interrelate the various users and to upgrade the hardware/software, data managment and personnel. A review of stages 4 and 2 vertically in Illustration 1 (Slide 4) demonstrates this point. The framework we have constructed not only helps to see and understand the present environment, but to put the problems of the present and associated consequences for the future into perspective.

CONCLUSION

While the primary purpose of this paper is not that of providing advice on how to traverse the evolutionary continuua, I am not going to completely resist the temptation to do so. The steps that follow may describe a general way to apply the continuua concept to a practical situation.

These are some of the evolutionary steps to take. They are not to be taken in strict sequential order. Some orchestration is required:

Refer to Slide 16: Summary:

1. Stabilize all existing operational level systems -- make them self sufficient using smallest possible amount of resource.

2. Reallocate resources needed by step 1. Obtain sufficient new resources to support the transition effort and prepare a permanent structure.

3. Reorient management: ·Reorganize
   - Education
   - New Functions
   - Management must be involved in determining strategy.

4. Reorient Users:
   - Education
   - Involvement
   - Internal changes in organization

5. Establish DBA and data dictionary :

·Hardware ·Software ·Human	Resource/Support

6. Choose DBMS

7. Shift all application to DBMS with appropriate computing resources, organizational support and user preparation.

We set out to develop a framework that would describe the evolution of administrative data processing, would delineate the role of data base technology in that evolution and that would be applicable to real-life situations. We found that the evolution has six major characteristics, which were described as continuua. The are:

Applications
Personnel Specialization
Management Techniques
Data Management
Hardware/Software
User Demand/Perception

We found that data base technology is playing a very key role in this evolution; however, the full extent to which data base is impacting us will only become clear as the future recedes into the past. We also found that our conceptual framework, when applied to a real-life case, does help us to understand problems of both the present and the future.

The usefulness of this framework lies in the ability to use it to provide some perspective on the mass of information, problems and demands that daily confront computing in the educational environment. This framework can be used to help us see not only where we have been, but where we are and where we are going.

1) Development of a framework to describe the ADP evolution toward Data Base technology.
2) An example of the framework applied to a real-life case, and
3) Some conclusions.

Figure 1

> ... Administrators generally have used computing facilities mainly in operational systems such as payroll and registration to process large quantities of data and repetitive transactions. A benefit also to be realized, however, is that accurate timely, and concise information can be provided to assist management in administrative decision - making and control.

(underscoring added)

Figure 2

Illustration 1

THE ADMINISTRATIVE DATA PROCESSING EVOLUTION

Time: 0 → Infinity

CHARACTERISTICS	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Future Stages
Applications	•Single function application program	•Application systems for discrete, operational level functions	•Moratorium	•Data based	•Total Enterprise	•Inter-Enterprise
Personnel Specialization	•Fundamental	•Increased for programmers	•Increased for maintenance & systems	•Increased for existing •New ones created	•Increased for all	
Management Techniques	•Lax •EDP under another Department	•Still loose •EDP recognized	•Separate Department •Control	•Resource oriented	•Service & planning oriented	
Data Management	•Elementary	•Sequential Records & Files	•Access methods •Redundancy	•Data bases	•Application Independent Data Bases	
Hardware/Software	•Elementary	•Sequential	•Bigger •Faster •Complex Operating systems	•Faster •Cheaper •Smaller •Software/ Firmware	•Families of machines	
User Demand/Perception	•Limited demand •Compartmentalized	•Demand for information increases	•Increasing Demand •Interrelated users	•Sophisticated reporting •Dependent	•Highly dependent	•Computers and people

FIGURE 3.

Illustration 2

APPLICATIONS CONTINUUM

Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Future Stages
1. Cost-reduction accounting applications 2. Single program 3. Examples: Payroll Accounts Receivable Accounts Payable Billing	1. Proliferation of applications in all functional areas &operational levels. 2. Functions discretely defined. 3. High degree of redundant program code 4. Predominantly serially executable 5. Examples: Cash flow General ledger Budgeting Capital Budgeting Forecasting Personnel Inventory Order processing Sales Inventory Control	1. Moratorium on new applications: emphasis on control. 2. Begin move toward information processing systems. 3. Reduction of redundancy increased parallel processing 4. Examples: Purchasing control Scheduling	1. Data-base applications 2. Interrelating of sub-systems 3. Continued reduction of redundancy & increased parallel processing 4. More emphasis on data organization 5. Supra-operational level orientation 6. Examples: Simulation models Financial Planning Models On-line personnel query system On-line source data entry (e.g., cost collection order entry)	1. Total enterprise information system 2. Keyed to support vertical & horizontal information flow based on key tasks/key events 3. Minimal redundancy of processing & data	Inter-enterprise

FIGURE 4.

Illustration 3

GROWTH OF PERSONNEL SPECIALIZATION CONTINUUM

Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Future Stages
Specialization for computer efficiency	Specialization to develop variety of applications	Specialization for control and effectiveness assurance	Specialization for data-base technology and teleprocessing		

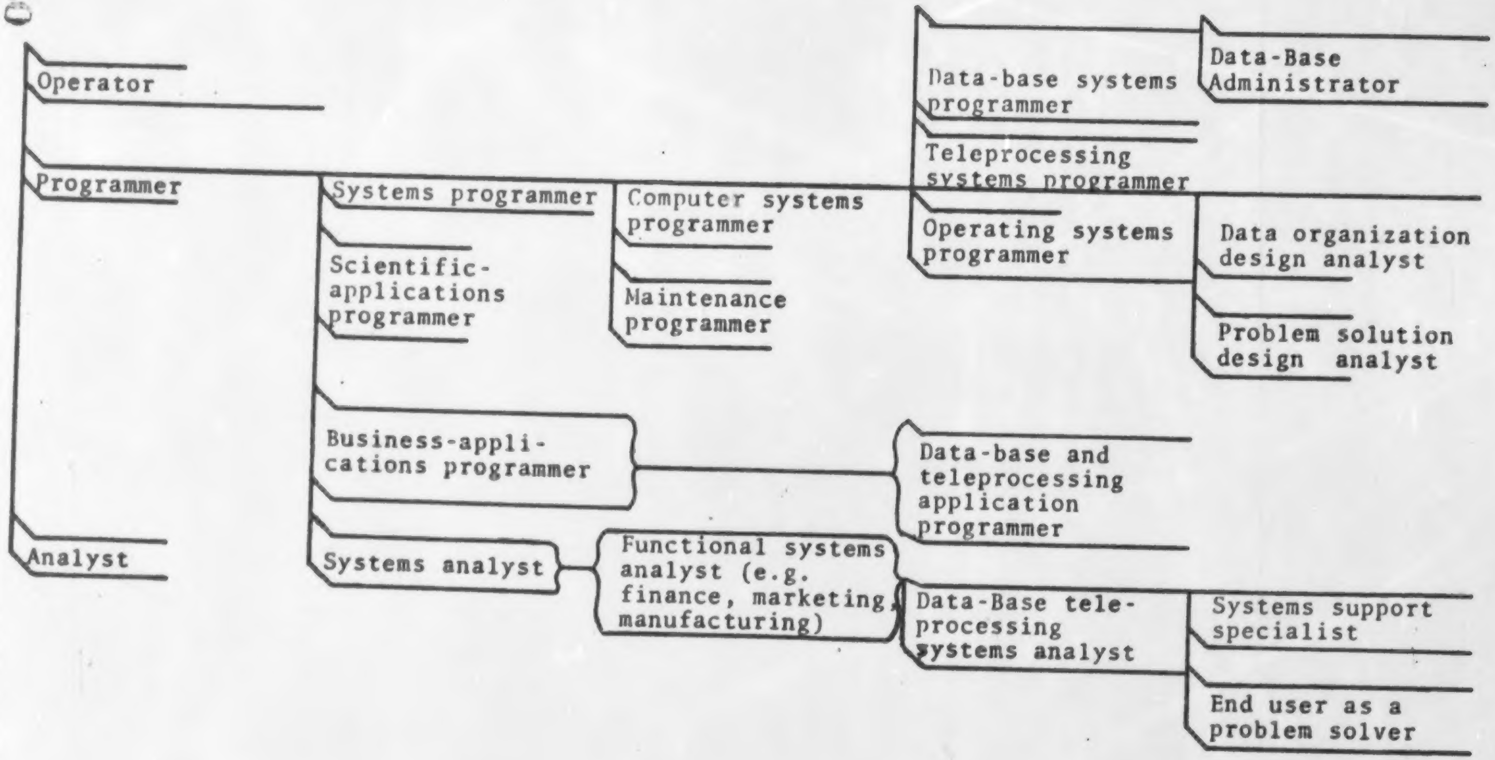

Operator
Programmer
Analyst
Systems programmer
Scientific-applications programmer
Business-applications programmer
Systems analyst
Computer systems programmer
Maintenance programmer
Functional systems analyst (e.g. finance, marketing, manufacturing)
Data-base systems programmer
Teleprocessing systems programmer
Operating systems programmer
Data-base and teleprocessing application programmer
Data-Base teleprocessing systems analyst
Data-Base Administrator
Data organization design analyst
Problem solution design analyst
Systems support specialist
End user as a problem solver

FIGURE 5.

Illustration 4 - C

MANAGEMENT TECHNIQUES CUSTOMARILY APPLIED IN EACH OF THE FIVE STAGES -- PLANNING

Stage 1 Lax management	Stage 2 Sales-oriented management	Stage 3 Control-oriented management	Stage 4 Resource-oriented planning & control	Stage 5 Service-oriented management, planning and control
Loose budget	Loose budget	Strong budgetary planning for hardware facilities and new applications.	Multiple 3-5 year plans for hardware, facilities, personnel, and new applications.	Planning and EDP become more integrated.

FIGURE 6.

Illustration 4 - B

MANAGEMENT TECHNIQUES CUSTOMARILY APPLIED IN EACH OF THE FIVE STAGES -- CONTROL

Stage 1 Lax management	Stage 2 Sales-oriented management	Stage 3 Control-oriented management	Stage 4 Resource-oriented planning & control	Stage 5 Service-oriented management, planning and control
Controls notably lacking; priorities assigned by FIFO; no chargeout	Lax controls, intended to engender applications development; few standards, informal project control.	Proliferation of controls to a runaway budget; formal priority setting; budget justification. Programming controls; documentation standards. Project management initiated; management reporting system introduced; project plan, project performance, customer service, personnel resources, equipment resources, budget performance. Chargeout introduced; postsystem audits. Quality control policies for computer system, systems design, programming, operations.	Refinement of management control system -- elimination of ineffective control techniques and further development of others; introduction of data-base policies and standards; focus on pricing of computer services for engendering effective us of the computer.	Management and control is diffused-major effort is to provide central coordination of the more decentralized EDP organization.

FIGURE 7.

Illustration 4 - A

MANAGEMENT TECHNIQUES CUSTOMARILY APPLIED IN EACH OF THE FIVE STAGES---ORGANIZATION

Stage 1 Lax management	Stage 2 Sales-oriented management	Stage 3 Control-oriented management	Stage 4 Resource-oriented planning & control	Stage 5 Service-oriented management, planning and control
EDP is organized under the department of first-applications. Justification: it is generally a small department	The EDP manager is moved up in the organization systems analysts and programmers are assigned to work in the various functional areas.	EDP moves out of the functional area of first applications; a steering committee is set up; control is exerted through centralization; maintenance programming and and systems programming become dominant activities.	EDP is set up as separate functional areas, the EDP manager taking on a higher-level position; sometimes programmers are decentralized to user areas; high specialization appears in computer configuration and operation; systems design and programming take on a consulting role.	EDP is distributed (networks, subsidary organizations)

FIGURE 8.

Illustration 5

DATA MANAGEMENT CONTINUUM

Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Future Stages
Data element is unit of data management. Records and files are very simple, sequential structures	Records and files sequential	Records and files but more complex forms of data organization, access methods and improved maintenance techniques; rudimentary"data bases", still tape oriented in design. Data redundancy is prevalent.	Data bases, application programs still need to know data structures. Data redundancy begins to be greatly reduced. New kinds of data elements are introduced, such as virtual data items, which can be calculated from other statistics stored internally.	Data, rather than programs, become the center of the application universe. Data management functions are completely separated from the application program.	Interrelated data banks moving toward comprehensive data banks.

FIGURE 9.

Illustration 6

USER DEMAND/PERCEPTION CONTINUUM

Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Future Stages
Small, finite group of users; limited scope or purpose for computing.	Various, finite groups of users; broader scope for computing, serve many operational needs.	Various, finite groups of users and interrelationships among them supported by computing at operational level with some computing support for management level.	Increased interrelationship of groups of users supported by management control and planning systems as well as more comprehensive information systems.	Computing support for the decision-making and related operational support at all levels within the enterprise.	Linking enterprise together, comprehensive linkages of enterprises.
User wants simple listings of data. Is tolerant of delays/problems.	User wants transaction data and summary data of various types. Is less tolerant of problems, is more dependent on computer for day-to-day activity.		User wants all preceeding forms of reporting but also wants profile and analysis data as well as simulation. Is very dependent on computer and wants it to be available always.	User wants computer to provide information his way, does not want to permit technical problems to limit him. Is totally dependent on computer and requires total reliability and accessability.	Computers are a fact of daily living. Computers may even be perceived as "users" to each other.

FIGURE 10.

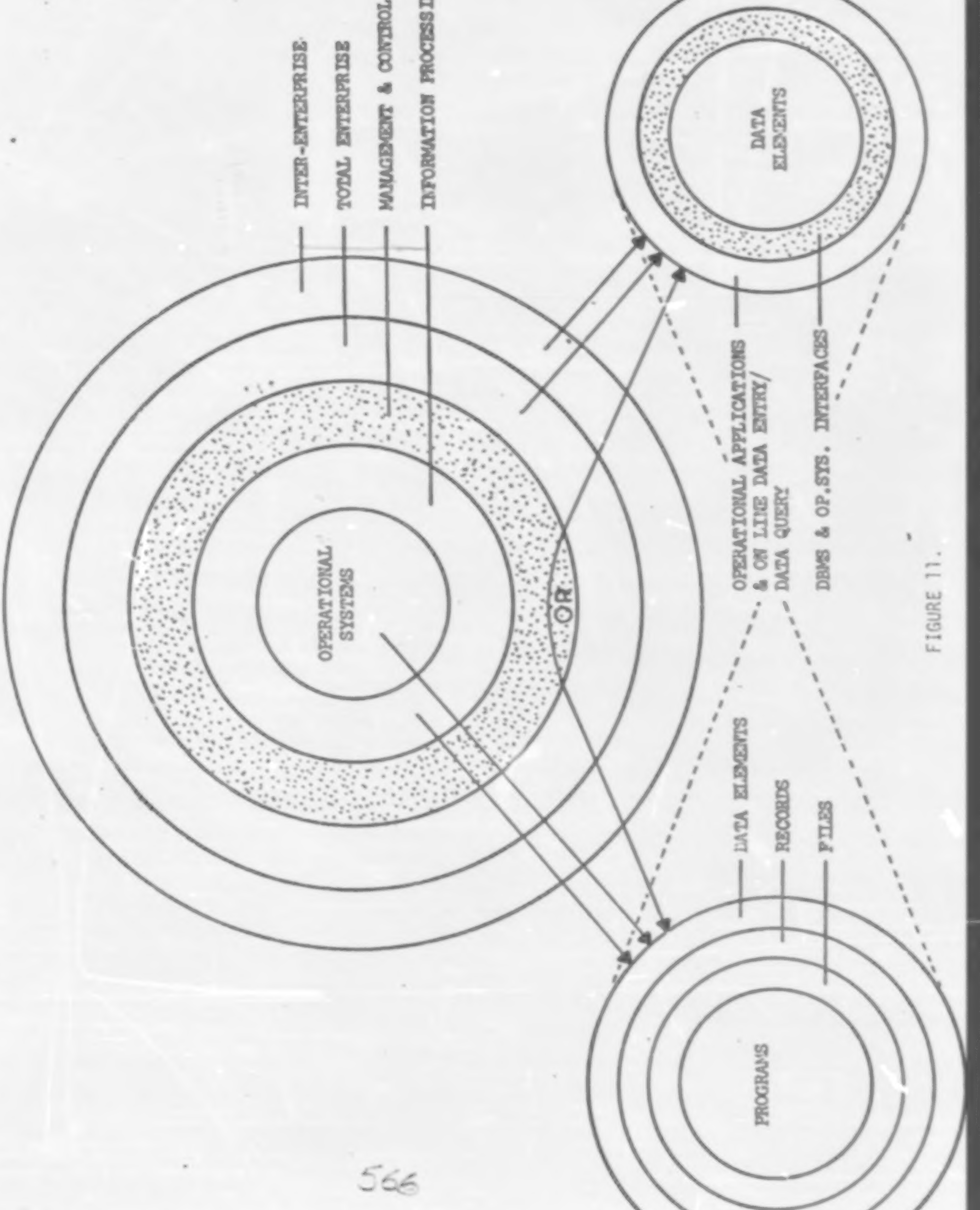

INTER-ENTERPRISE
TOTAL ENTERPRISE
MANAGEMENT & CONTROL
INFORMATION PROCESSING
OPERATIONAL SYSTEMS
OR
DATA ELEMENTS
OPERATIONAL APPLICATIONS & ON LINE DATA ENTRY/ DATA QUERY
DBMS & OP.SYS. INTERFACES
PROGRAMS
DATA ELEMENTS
RECORDS
FILES

FIGURE 11.

THE COMPUTERIZED INFORMATION OUTPUT ONION

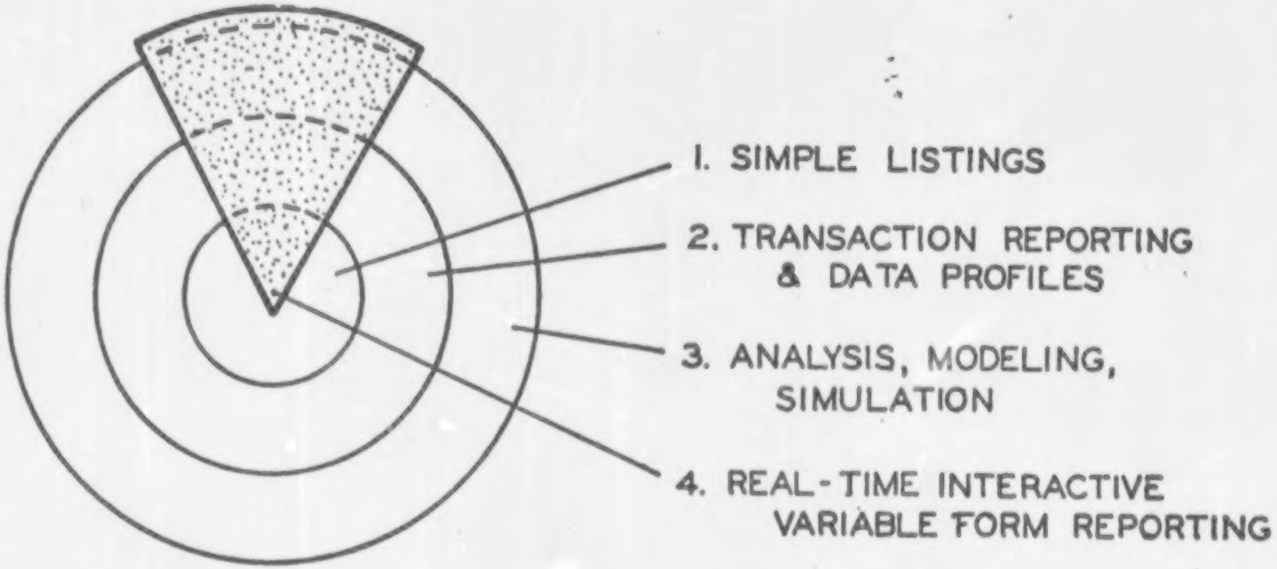

1. SIMPLE LISTINGS
2. TRANSACTION REPORTING & DATA PROFILES
3. ANALYSIS, MODELING, SIMULATION
4. REAL-TIME INTERACTIVE VARIABLE FORM REPORTING

FIGURE 12.

Illustration 1

THE ADMINISTRATIVE DATA PROCESSING EVOLUTION

Time: 0 → Infinity

CHARACTERISTICS	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Future Stages
Applications	•Single function application program	•Application systems for discrete, operational level functions	•Moratorium	•Data based	•Total Enterprise	•Inter-Enterprise
Personnel Specialization	•Fundamental	•Increased for programmers	•Increased for maintenance & systems	•Increased for existing •New ones created	•Increased for all	
Management Techniques	•Lax •EDP under another Department	•Still loose •EDP recognized	•Separate Department •Control	•Resource oriented	•Service & planning oriented	
Data Management	•Elementary	•Sequential Records & Files	•Access methods •Redundancy	•Data bases	•Application Independent Data Bases	
Hardware/Software	•Elementary	•Sequential	•Bigger •Faster •Complex Operating systems	•Faster •Cheaper •Smaller •Software/Firmware	•Families of machines	
User Demand/Perception	•Limited demand •Compartmentalized	•Demand for information increases	•Increasing Demand •Interrelated users	•Sophisticated reporting •Dependent	•Highly dependent	•Computers and people

FIGURE 13.

Illustration 9

EIS FIVE YEAR PLAN -- A.D.P. EVOLUTIONARY STAGES

YEAR 1	YEAR 2	YEAR 3-4-5-	YEAR +5
Limted number of application system available Stabilized support and reliability of existing applications.	Expansion- Rapid development of new applications aimed primarily at providing computer support for administrative functions that were entirely manual. Most development done by importing and modifying software packages.	Integration of data element. Move to IMS or IMS-like system. Movc existing applications to "IMS" environment. Stabilize. Add modest amounts of new application software	Full Data Base planning Simulation Distributed computing

2 application systems

11 application systems

Stable level of application availability;emphasis on improved support, reliability, distribution of function, expansion of type of user and kind of use.

FIGURE 14.

- STABILIZE
- REALLOCATE RESOURCES
- REORIENT MANAGEMENT
- REORIENT USERS
- CHOOSE DBMS
- CONVERT TO DBMS

FIGURE 15.

References:

(1) UNI-COLL Newsletter, Vol. 11, No. 9, June 1974, University City Science Center, Philadelphia, Pa. 19104.

(2) Data Processing in 1980-85: A Study of Potential Limitations to Progress, (also called SILT Report), SHARE, Inc., 1975.

(3) Gibson, Cyrus F. and Nolan, Richard L., Managing the Four Stages of EDP Growth, Harvard Business Review, January - February, 1974.

(4) Bachman, Charles W., The Programmer as Navigator, Communications of the ACM, Vol. 16, No. 11, November 1973.

(5) Gallitano, Joanne M., Administrative Data Processing Goals and Objectives, in Educational Information Services, Inc., FIVE YEAR PLAN, New Brunswick, New Jersey, February 1973.

(6) Nolan, Richard L., Computer Data Bases: The Future Is Now, Harvard Business Review, September - October 1973.

(7) Withington, Frederic G., Crystal Balling: Trends in EDP Management, Infosystems, January 1973.

(8) Goggin, William C., How Multidimensional Structure Works at Dow Corning, Harvard Business Review, January - February 1974.

(9) Jardine, D.A., A Tutorial on Data Base Architecture, unpublished.

(10) Soden, John V., Planning for the Computer Services Spin-Out, Harvard Business Review, September - October 1972.

(11) Sibley, Edgar H. and Turner, Jon A., Data Base Management: A Framework for Effective Use, Proceedings 2nd Jerusalem Information Technology, August 1974

(12) Dearden, Myth of Real-Time Management Information, Harvard Business Review, May - June 1966.

THE TREND TOWARD A COMMON DATA BASE: LANGUAGES AND STRUCTURES

Mike O'Connell
Software Project Manager
Business Products Group
Digital Equipment Corporation

THE TREND TOWARD A COMMON DATA BASE - LANGUAGES AND STRUCTURES

Mike O'Connell
Digital Equipment Corporation

There is a definite trend toward a common set of data base languages and data base structures, similar to the trend toward common programming languages in the 1960s. CODASYL, with participation by all the major computer manufacturers, government agencies, and users, has been developing an industry-common collection of data base languages and structures. Now, nine years after beginning, their work has resulted in an increasing number of implementations available on a wide variety of computers.

CODASYL has proposed three types of languages:

- A Data Description Language, used by the Data Base Administrator, to describe the content and structure of the data base, that is, to define the "Schema"
- A Data Base Description Language, used by the Data Base Administrator, to select that portion of the data base to be accessible by a given application program, that is, to define "Sub-schemas"
- Data Base Manipulation Languages, used by the application programmer, which are extensions to existing programming languages to provide access to the data base.

The data base structure proposed is based on the concept of the "set", which is simply a named collection of records that bear some form of relationship to each other. When defining the types of records that participate in a given type of set, the Data Base Administrator may choose from among all the types of records in the data base, without restriction. Thus, a specific type of record may participate in any number of different sets. By naming the same record type in different sets, more complex structures may be built, such as chains, rings, trees, and networks. The advantage of CODASYL's approach, contrasted with other approaches, is that the resulting structures in a single data base can be as simple or as complex as the user needs.

The current status of CODASYL data base activity is:

- The Schema language has been approved by CODASYL and is a candidate for standardization whenever ANSI establishes a technical committee to do such work.
- The Sub-schema language and the base base manipulation language for COBOL are expected to be approved early in 1975. Then they will be considered for standardization in the same process previously used by ANSI to standardize COBOL.
- The data base manipulation language for FORTRAN is under development and is expected to be completed late in 1975. Again, ANSI will use its existing procedures for further versions of ANS FORTRAN.

TRACK IV

Hardware and Systems Software

Gale R. Aguilar

Earl C. Joseph

Jack E. Shemer

Track Coordinator: John Gwynn

DIRECTIONS IN SYSTEM DEVELOPMENT: KEY TO ENHANCED PRODUCTIVITY

Gale R. Aguilar
Director
Systems Marketing Programs
IBM Corporation

DIRECTIONS IN SYSTEM DEVELOPMENT: KEY TO ENHANCED PRODUCTIVITY

Gale R. Aguilar
Director
Systems Marketing Programs
IBM Corporation

Mr. Aguilar traced the history of systems development, emphasizing the evolution of System 370. While changes have been gradual, today's technology represents a significant advancement in the state-of-the-art. Looking at ways to use this technology productively and effectively, Mr. Aguilar stressed interactive applications and data base communications.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

FUTURE COMPUTERIZED EDUCATIONAL SYSTEMS

Earl C. Joseph
Staff Scientist
Sperry UNIVAC

FUTURE COMPUTERIZED EDUCATIONAL SYSTEMS

Earl C. Joseph
Staff Scientist
Sperry UNIVAC

Through a scenario, based on our current visibility of computer unication trends, a possible future for education and society was discussed and based on the rationals for humanistically shaping the future. Thus far, this decade is, in many ways, a pivotal point in the computer field resulting in placing into clearer focus a number of definite trends that point to the dominate technology, architecture, special features and application of future computers. This presentation discussed these turns, presented the resulting base line trends and made an extrapolated prognosis for the next decade and beyond. Any serious investigation of future computers, from extrapolations of current advancements, now point to the important questions, "What's beyond computers?" Especially, when the agenda of possible futures are mapped, many possibilities exist. Thus, being at the brink of a new computer revolution in the makeup of computers this presentation discussed the more important trends that pointed to the windows of opportunities for future education resulting advances from computers.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

IMPLEMENTING STRUCTURED PROGRAMMING

Donald H. Bein
General Manager
Data Systems Division
A. O. Smith Corporation

IMPLEMENTING STRUCTURED PROGRAMMING

Donald H. Bein, General Manager
Data Systems Division
A. O. Smith Corporation
P. O. Box 584, Milwaukee, Wisconsin 53201

A. O. Smith's Data Systems Division, in the pursuit of greater flexibility and lower cost for its information systems, has been practicing structured programming techniques for the past two years. The presentation will show how to do structured programming, will discuss some of the subtle problems that arise, and will report on the benefits that are being achieved.

1. ENVIRONMENT

A. O. Smith Corporation is a diversified manufacturing firm celebrating its Centennial this year. The smallest of its autonomous divisions is the Data Systems Division that is now in its 14th year of selling its computer services to other divisions of A. O. Smith and to other firms as well. With two inputs of equipment and manpower, this organization has two essential outputs: 1) reliable, high-performing computing; and 2) new information systems. We want to talk about how we have implemented structured programming in the development of new information systems, and we should start with our major objectives. There are three: 1) No more big conversions; 2) More flexibility (easier change) in our systems; and 3) Lower development costs.

2. THE OBJECT IS COST REDUCTION

The high costs of information systems development are accentuated by the ever-rising proportion of the computer dollar that is spent for software as opposed to machines. Eighteen years ago A. O. Smith had ten people programming its $30,000 per month 705. Today it has 120 people programming its $120,000 per month System 370/165. While machine costs increased four times, software costs have increased 36 times!

A. O. Smith's Data Systems Division is tackling this problem of high system development cost with a three-pronged strategy: 1) Generalized Systems; 2) Data Base Technology; and 3) Structured Programming.

Very significant accomplishments have been made in reducing development costs with all three approaches. Perhaps most significant is the generalized systems which are designed to operate for no one but are customizeable to fit almost anyone. Last year one of our factories spent $17,000 to obtain the functional equivalent of an inventory control system for which ten years ago another plant spent $500,000! In fact, it is these powerful generalized systems in manufacturing, accounting, and engineering that form the basis of our very successful business venture in selling computer services to other companies.

Our work with data base technology is also extremely promising. But this paper will report on our implementation of structured programming. The results that are in are sparse as yet, but there is no doubt that structured programming promises to be a major contributor to information system development cost reduction.

3. OVERVIEW

In this paper we will point out the difference between modular and structured programs, speak to the important topic of programming style, outline "how to do" structured programming, and point out both problems and advantages.

4. MODULAR PROGRAMMING DISAPPOINTMENTS

Modular programming is what we asked for ten years ago. What we got was modular all right, but it just didn't deliver the hoped for cost reductions. We received modules instead of programs; and instead of 12,000 statements, they were only 1,000 statements. You could carry the listing. These modules were still much too large for easy debugging and maintenance. They were functionally-oriented at a very gross level such as edit, update, report writer. Our "one pass of the master tape" heritage was working against us as we tried to squeeze as much processing power as we could into each module. And these modular programs were not structured. A good example of what is meant here is IBM's Operating System -- highly modular indeed, but so populated with inter-dependencies that fixes and changes changes became monumental efforts. Another example is the typical FORTRAN application -- highly modular but every module refers to COMMON in a way that again we have one unstructured mountain of inter-dependencies. We asked for modular programming, but we failed to give our programmers a uniform set of constructs with which to build programs in a structured way. They were designed on the coding pad and we often had the inverted project life cycle: We tested until the programs were coded, and we coded until the programs were designed.

5. STRUCTURED PROGRAMMING

5.1 Design Approach. And that concept of design is the most important aspect of what we call structured programming. This design approach is one that must be practiced from the top down. The entire information systems must be considered with firm definition of what the system is to do. HIPO (Hierarchical Input-Process-Output) diagrams are very useful to show how information flows. The user must be involved in the creation and the critique of these HIPO diagrams. The user agrees to each lower level of detail as the definition of what the system is to do.

Once the "what" has been thus defined in hierarchical fashion, the programming team is ready to begin on the "how". Again, this is at first kept strictly at the system level with no regard to computer programs. Out of this analysis of what gets done and how it might get done comes logical clusters of functions. These clusters lead to the specification of computer programs. These specs will spell out the function that is to be performed, input, output, and constraints.

In proceeding to the next lower level of program design, the important aspect here is that program structures and data structures are considered together. Putting the code together properly eliminates intermediate data and appropriate data structure eliminates code. The two must be done together.

5.2 Data Structure Design. The data structures are three: 1) the data base which is primary and permanent; 2) the in-core processing, where careful attention to structures can vastly simplify the code that is written; and

3) the intermediate files if, and only if, any are necessary. That is why they are to be considered last -- or else we may find code written to process intermediate files that are not really necessary!

5.3 Program Structure Design. Program structures should be thought of as two kinds: 1) processing modules that mirror the practical computer application with a tree structure, and 2) functional service modules that may be thought of as the black boxes that appear many places in a system with a network calling structure. Examples of processing modules are time ticket edit, FICA tax calculation, etc. Examples of service modules are error routines and complex number transformations.

5.4 Sketch Coding. Once a hierarchy of processing modules and service modules are defined, they should be sketch coded in an abbreviated pseudocode of part English and part programming language that can easily be read by one's colleagues.

Now "egoless" programming takes over as a programmer welcomes criticism while the program is still in the sketch stage.

5.5 Only Three Constructs. Here is where a discipline of using only three programming constructs is applied: 1) the sequential procedure; 2) looping (DO WHILE ...); and 3) the conditional branch (IF ... THEN ... ELSE ...). There is one way in and one way out of each construct. GO TO is quite absent.

5.6 High Modularity. A discipline of high modularity is also applied at this time. A goal of 30 or 40 executable statements per module is strived for with an absolute maximum of 120 statements. This sketch coding easily shows if these requirements will be met once actual coding is commenced. If modules are too large, they are broken down still further into smaller but well bounded modules of limited function.

6. PROGRAMMING STYLE

Programming style is a most important part of structured programming, and it is revealed with this sketch coding. Programming style is much more than the limitation of only three constructs. We want a programming style that can be likened to the style of an architect as contrasted to that of an artist. No one will deny that creative genius can be an important part of the attributes of a good architect. However, he uses his genius not just for elegance in appearance but for practical functionality and cost considerations as well.

6.1 Linearity -- No GO TO's. The elements of style that we strive for are first of all linearity with the absence of GO TO. Aided by the formatting capabilities of the PL/I checkout compiler, an indented sequential listing is very readable. The listing is both flow chart and listing at once.

6.2 One Page "Chunks". We also strive for locality with modules of executable logic (exclusive of data definition) to be no more than one page of code. There should no longer be the constant flipping of pages of the listing as a programmer debugs his work. Elements of logic are brought together in a structure of nested statements such as "DO WHILE" and "IF ... THEN DO ... ELSE IF ...". This creates "chunks" of logic that can usually be nicely placed on one page. This "chunking" is in considerable contrast with the flow charting conventions that jump around to page after page of documentation manuals.

6.3 Meaningful Identifiers. We try to use meaningful names for data elements so as to make the programs self-documenting as much as possible. With meaningful identifiers, comments can be sparse for there is no need to echo the code with comments that in all probability will never be changed as the code is changed in debugging and maintenance.

6.4 Coding Organization. A final point in programming style is to strive for a discipline of code organization. Not only are modules clearly defined as processing and service modules, but each module has its own organization where data description is kept quite separate from the logical procedures. Chunks of this logic are are clearly set off by proper indentation. Comments are used only for a general description of the logic and the data.

6.5 Stress on Discipline. Add this all up and we see in structured programming the continual emphasis on a design discipline. It starts with a top-down approach to a computer application and ends up with a program listing that is readable, self-documenting, and ready to welcome change. How all this has been applied in practice will now be described with four case studies.

7. CASE STUDIES

7.1 Posting Project. The first is a work-in-process inventory posting project. This project started as an expensive proposal to patch an existing program, adding many new rules for posting very complex open production orders. By the time the customer decided to purchase our proposed patch, we bravely decided to use his funds instead to rewrite the entire system going full boat using all that we knew about our new structured programming techniques. The job was completed in 12 man-months -- the same cost as estimated for the patch.

7.2 Estimating Project. In a cost estimating project, structured programming was used and costs on this project were over their estimate, although this was largely because of new data base technology being introduced. The post mortem criticized the "egoless" reviews as being "far too picky" and costly. The quality of the code was excellent, as the user experienced no program failures.

7.3 Steel Coil Project. In a project for steel coil inventory control, the "egoless" review concept was concentrated in the sketch code phase with success. Despite some problems in defining modules properly, this major project was completed with considerably less cost and elapsed time than has been normal for this very large and complex customer.

7.4 Budget Project. A budget system is still being finished, and structured programming took some knocks as personality conflicts gave us less than the smooth working programming team that the technique calls for.

8. SUBTLE PROBLEMS

8.1 "Egoless" Environment. This leads into a discussion of some of the subtle problems that arise as an organization implements structured programming. We have learned that "egoless" programming by no means comes as naturally as one might suspect. The concept must be promoted. Again, the artist who says "here is my work; if you don't like it -- tough!" must be contrasted with the scientist who says "here is my work; try to tear it apart!" This concept as a means to the end result of lower cost, higher quality programs must be sold. It does not come naturally.

A certain amount of tact that keeps professional objectivity uppermost is a necessity in all review sessions. "Cheap shots" must be prohibited, and honest criticism cannot be discarded under a "cheap shot" label. For this reason alone, management of the programming effort must stay intimately involved. Delegating structured programming to the technicians to implement will most certainly result in failure.

8.2 New Documentation. The new documentation -- HIPO (hierarchical input-process-output) charts -- take some getting used to. Proficiency grows with use. The lack of flow charts is a bigger problem. Once people see that an indented listing really contains all the information that's on a flow chart, they can change. But some people who grew up with flow charts may always need them as a preliminary to sketch coding. That doesn't matter if the final result is good, well-structured, and self-documenting final code.

8.3 Designing is Work. Design -- the substance of structured programming -- is hard work! The tendency of eager programmers is to rush to the coding pad. Coding is tangible -- designing is abstract. Considering coding structures and data structures simultaneously is not a simple matter. And when the techniques are all new, the hard work is even harder! Hence we find the final stages of a project don't get the benefits of the same diligence applied to the earlier phases. Again management must stay involved to apply the same discipline even-handedly through all phases of a project.

9. SUMMARY

In summary, despite the problems of introducing new technology, there is no question but that structured programming is promising. Development costs should be less, and maintenance costs considerably less. There is the promise of a higher quality product with less errors. Programs that are readable and understandable are easy to change. Our customers have been demanding increased flexibility. Data base technology is already delivering the goods here -- now structured programming promises to make its contribution.

Most importantly, structured programming offers the biggest promise in managing the increased complexity that we all face. The days of simple data processing are past. The computer today is truly an information processing machine. The complexity of the jobs it tackles must necessarily grow as it moves increasingly into the operating areas of business. Our old methods just won't handle these larger problems. For the computer to fulfill its destiny in society, these large problems must be decomposed down to manageable size. And right now we see one way of doing this -- STRUCTURED PROGRAMMING!

THE REALITIES OF THE USED COMPUTER AND THIRD-PARTY LEASE MARKET

George Spanos
Vice President
Data Options Corporation

THE REALITIES OF THE USED COMPUTER AND THIRD-PARTY LEASE MARKET

George Spanos
Vice President
Data Options Corporation

This presentation outlined the benefits and advantages of dealing in the used computer and third-party lease market, and how to deal with the implied innuendo of the various manufacturers, and their representatives which sometimes occur. Typical sales were discussed with audience participation.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

DISTRIBUTED PROCESSING SYSTEMS

Dr. David J. Farber
Professor
Information & Computer Science
University of California, Irvine

DISTRIBUTED PROCESSING SYSTEMS

In the past several years a notable trend has developed in the use of minicomputers. We have seen the formation of a number of these computers into distributed function systems. Dr. Farber discussed the economic, technical, and environmental reasons which underlie this trend. One such system was examined as an example...the DCS System developed at the University of California at Irvine under NSF sponsorship. Future developments and their impact were then projected.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

REALITY, A MICRO-PROGRAMMED DATA MANAGEMENT SYSTEM

Richard Pick
President
Pick & Associates, Inc.

REALITY, A MICRO-PROGRAMMED DATA MANAGEMENT SYSTEM

Richard Pick
President
Pick & Associates, Inc.

REALITY includes hardware and software and has been specifically designed to provide a tool for the rapid and effective implementation of on-line, transaction oriented data base management systems. The REALITY software includes ENGLISH, a data management language. ENGLISH was derived from public domain software and was the starting point for all subsequent hardware and software development. The architecture of the REALITY processor; the micro-programmed, multi-user, virtual memory operating system; and the supporting system software were all developed subsequent to ENGLISH. This approach has resulted in a generalized data management facility certainly equal in capability and throughput to those currently available on large (i.e. expensive) systems, but at the same time, the REALITY hardware is a low-cost mini-computer. REALITY was designed and developed by Pick & Associates and is manufactured and marketed by Microdata Corporation. Numerous users have used REALITY to implement integrated data management systems.

The abstract of this paper is included for the interest of the reader. Unfortunately, the full paper was not received in time for publication. For more information, please contact the author.

MINI-MAXI MIXING: A HIERARCHY FOR ACADEMIC COMPUTING

John H. Esbin
Acting Director
Computer Center
University of Iowa

MINI-MAXI MIXING: A HIERARCHY FOR ACADEMIC COMPUTING

John H. Esbin, Acting Director
Computer Center, University of Iowa
Iowa City, Iowa 52242

The value of interactive terminals is now well established, both for academic and administrative purposes. However, to provide a full range of computing services through interactive terminals to large numbers of users requires either massive general purpose computers or innovative combinations of more specialized systems. A hierarchial system is under development at the University of Iowa which will provide the timesharing services of mini-computer based systems, but will also allow access to a large-scale general purpose system for more sophisticated applications. Primarily intended for instructional use, the hierarchy is also adaptable to research and administrative uses. This presentation reviews the background and progress of the development, with projections for future growth and applications.

1. INTRODUCTION

Higher education has as its primary goal the education of students who pass through its programs. The graduates of higher education have made an impact that is inescapable in our society, yet we are faced with criticism on many fronts. Perhaps some of this is deserved, and certainly some of it is not, but there is no institution of higher education that can claim perfection. No school ever prepares its graduates for everything, or has the most efficient administration conceivable, or has taken advantage of all the technology that has been developed in the history of the world.

Of all of the developments since the technological revolution none have been as important as the development of the computer. While some may say that this is a moot point, increasingly the computer appears in every aspect of our lives. The computer is perhaps the center of the whole field of educational technology, and yet it has not been sufficiently exploited to take advantage of its many features and capabilities. Since criticism on this score is valid, the aim of this paper is to explore one general way in which a specific application of educational technology may help to change the current situation. What we will deal with is the question of machines, systems, and people utilized in new ways to do what universities should do, namely educate and add to the body of knowledge.

2. A FRAME OF REFERENCE

An eloquent report on this subject has been produced by the Carnegie Commission, entitled The Fourth Revolution.* We subscribe to the concepts which that report suggests:

1. The quality of teaching, which varies now depending upon the excellence of the teacher, can be uniformly

---

* The Fourth Revolution: Instructional Technology in Higher Education. The Carnegie Commission on Higher Education, McGraw-Hill Book Company, New York, June 1972.

upgraded by the technology which is now or will be available in the next two decades.

2. The student will be presented the opportunity, if not the mandate, to self-pace his learning, enriching his experience largely by the challenging material available to him on an individualized basis.
3. Learning will not be lock-step. Far from dehumanizing education, technology will allow the student to adjust his education to his own peculiar profile.
4. With equipment provided to allow the student to acquire much of his learning experience, the instructor will be freer to carry on individual and small group work with the students.
5. With our society increasingly dependent on the material goods produced by technology, study involving technology will, if properly understood, allow the student to determine in his mind the role of man in this world.

We dwell in this paper on the central fact of educational technology, the computer, and the role it should (and hopefully will) have at the University of Iowa.

3. COMPUTING IN HIGHER EDUCATION

3.1 General Uses Computers are devices which process information. The stereotype of computing is that of a mathematician happily employing the computer to calculate formidable formulas on excruciatingly long numbers to produce results understood by him and a handful of colleagues scattered around the world. And of course some of this takes place. But it is essential to realize that although much of the work done by a computer is numeric, an at least equal amount of the academic usefulness of the computer is due to its handling of non-numeric material. Some perhaps work-a-day illustrations: authorship of disputed literary material determined at least in part by computer examinations of textual material; translation of languages by computer; deductions on social and political issues based upon trends revealed by study of historical data; diagnostic assistance to physicians of patient ills; auxiliary, pre-screening of patient history; insight into artistic technique, both visual and aural, with the attendant creation of new art; literature searches to assist the already deluged scholar to pinpoint material vital to his further research. But perhaps the paramount usage of the computer in a non-numeric way is the creation and use of large data bases coupled with ways to extract relevant material for a broad spectrum of improved understandings.

What are the ways that computing impinges on education today? In a number of ways, which may be categorized into direct and indirect. The former includes computing with an academic orientation, the latter, the support features such as administrative data processing. These may be further subdivided as follows:

3.1.1 Direct: Academic Orientation

Research use of a computer
Instructional use of a computer
Laboratory on-line computing
Information retrieval systems
Research and instruction about computers and their uses

3.1.2 Indirect: Administrative Processing

Business Office, Registrar, etc.
Hospital patient systems
Institutional research
Management information systems

Administrative data processing is not in the scope of this paper save to say that as much imagination in its implementation could be applied as in any other field. Where it should meet academic computing must be explored, and this issue must eventually be solved. To date, no clear direction exists based upon the experience of universities which have attempted various combinations of the academic and administrative computing needs of their institutions.

3.2 Importance to Instruction Just what about computing gives it such an imperative as the Fourth Revolution claims for it? A number of answers spring to mind, in no special order:

1. The computer is the only invention in man's history which is an intellect amplifier. The existence of the computer compels us to seek a division of labor, with man doing what man does well, and with the computer doing what it does well. Far from relegating man to an inferior position, the computer should allow us to reserve just the most creative and imaginative intellectual tasks for us.

Since the computer is a tool of such cutting impact, it behooves every student, and more necessarily every instructor, to begin to examine its uses in each field of human endeavor.

2. As far as research is concerned, no statement is needed to defend its importance. Simply observing how computer usage, principally for research, has grown at all American universities in the past fifteen years is enough to convince one that computing is intrinsic to research. It is fascinating to observe, however, that this usage is not restricted to the physical and biological sciences. Indeed, the social sciences probably make heavier use today than the "hard" sciences.

3. For the manipulation of large files of data, the computer is so obviously superior to humans that virtually every institution of any size has adopted it for its administrative systems. What is interesting about this aspect of computer use, however, is the certainty that the effectiveness of any automated data processing system is directly dependent upon the skill and ingenuity of the people who create it. This in itself has implications for research and instruction.

4. In the instructional process no simple answer is possible. Instructional usage of the computer is truly only in its infancy. But consider the promises:

a. When an instructor teaches a subject from year to year, or even from day to day, his presentation varies not so much because his ability varies, but because he's affected by the weather, or his love life, or his last meal, or for a host of other reasons. With computer augmented teaching, that part commited to the computer can be indefinitely polished, and when once good, can not get worse.

b. The student can be faced with real situations with real data. This is both motivational and a way of making academic exercises merge with post-collegiate reality.

c. To the degree that augmenting material has been committed to the computer, the student can be in a self-paced situation. The least we can expect is increased student enthusiasm; at best the number of years required for degree acquisition may be shortened.

d. To the degree that the augmenting material has been perfected, material can be infinitely individualized, so that far from the human being's losing his individuality, it can be superbly reinforced.

e. The student will be exposed to computers for all of the 50 odd years left to him after college days. Since college education is intended to parade past the student all that is best of our civilization, so as to enable him to cope with if not direct his future, and since computers simply have to be an ever-present phenomenon in the next decades, the student must be prepared to see computing vis-a-vis each field.

4. DEVELOPMENT OF ACADEMIC COMPUTING

4.1 Definitions Before examining the historical development and needs for computing we set forth some definitions of terms to be used in what follows.

Batch Processing. This mode of computing is such as to accept jobs to be run, loading them into a queue, and running each job only as the jobs received earlier or at a better priority level are completed. The time from submission of a job to its return to the user may vary from a few minutes to several hours to a day or more. Time of return will be known approximately at submission time. Input-output areas for job submission and retrieval may be either central to or remote from the computer.

Time-sharing or Interactive Processing. In this mode of computing, each user has direct communication with the computer by means of a terminal (e.g., typewriter or cathode ray tube terminal) and appropriate communications equipment. Each user is given a slice of time, and on a round robin basis, each user gets his time slice in turn. Response time, the time from completing one line of input until the computer signals an O.K. for another line, is usually on the order of 5 seconds or less.

Remote Job Entry. This system uses the equipment of a time sharing system, but simply allows input into a batch system.

Computer Managed Instruction (CMI). This is a special mode of computing with variations possible. At the lower end, a computer is used simply to help with class records, usually on a daily basis, with print-outs provided to the instructor. At more advanced levels, remedial work may be specified by the computer to assist the teacher in working with the student. This may be a batch system, a remote job entry system, or an interactive system.

Computer Assisted Instruction (CAI). There is no universally accepted meaning for this term, but it is used here to mean a system of computer use wherein a student interacts with a set of computer materials prepared by an instructor. The usage may be drill and practice; it may be quiz with remedial work; it may be tutorial usage, with the computer instructing the student. Generally all of these methods require the use of a fairly sophisticated master language, in which the instructor encodes his course material. The student generally needs to know nothing about computing or programming.

Computer Based Education (CBE). This is a broad term, signifying any use of the computer to further the education of a student. It can range from the adjunctive use through increasingly intense interactions between the student and the computer, up to tutorial CAI. It can range from batch usage through interactive usage.

4.2 Historical Development and Needs Since the earliest days of computing at universities the computer in the academic arena has been thought of nearly exclusively as a research tool. At our university, the situation is the same. Yet we are principally in the business of educating, and that principally at the undergraduate level. And precisely in that area, the University of Iowa, as with nearly all other universities, had made no real effort to promote the use of computing. For instructional processing we acknowledged that we were simply inadequate. With our present (albeit fair-sized) computer system and staff we were unable to mount a major drive to increase instructional usage for the reason that we know that our present system cannot sustain such usage.

For instructional computing we see a need for a really convenient, rapid-turnaround batch system, but more important, a truly adequate time-sharing system. The batch system would probably be best provided on some kind of distributive basis, with batch terminals at a number of strategic locations on campus. The disruption of the thought process must be minimized if we are going to optimize the learning ability of the student, so as far as time-sharing terminals are concerned, they must be provided in numbers and locations enough so that most students can gain access to them with a minimum of discouragement. Finally, for research in Computer Assisted Instruction, adequate terminals, computers, and CAI languages are necessary.

For research computing we see the need for a computer whose capacity is not so little more than its load that the ceiling discourages further development and use.

5. IOWA'S PLAN

5.1 Why Interactive Computing for Instructional Usage? A number of reasons can be presented for concentrating on interactive computing for instruction. Among these are:

1. The student can usually complete his entire assignment in one sitting, with numerous reruns to convert the minor errors which in a batch system would cause days of delay.
2. As a result the assignments made by instructors can be kept synchronous with the material being taught in class.
3. The encouragement which the student receives to experiment beyond mere assignments on an interactive system can cause his learning experience to be far richer than on a batch system.
4. The observation of real phenomena in close temporal proximity can cause students to make deductions which

the same results, spread over a week of time, seem to conceal.

5. The simplicity of terminal use as compared to batch allows the student to concentrate more on the substance than on the computing.
6. The sheer excitement of working at a terminal just does not seem to fade.

What promise does massive use of interactive computing hold for the instructor? Perhaps the most important fact, once he is convinced that computing could play a role in his teaching, would be a searching reappraisal of both his teaching content and method. At the very lowest level, the instructor could well assign problems relevant to his material, to be worked with real data by the students at terminals. As the instructor progresses in computer sophistication, he could construct computer dialogues with the student which with proper care could be tailored to determine and remedy each student's deficiencies. With such use, the instructor could well find he has time to dwell on coordinating and relating the diverse material of his course in a way not before possible. In the long run, more material could be covered in a given time, or the depth of understanding could be greatly improved. It is not unreasonable to suppose that a massive reversion to computer augmented teaching could reduce the study of the average college student at the university from eight to seven semesters or less.

Can this be accomplished with less exotic devices then a massive interactive system? The answer seems to be no. For witness the schools attempting to use ordinary computer methods in upgrading the classes (see for example, the proceedings of the First, Second, Third, Fourth, and Fifth Conferences on Computers in the Undergraduate Curricula, held at Iowa, Dartmouth, Atlanta, Claremont, and Wahsington State in 1970, 1971, 1972, 1973, and 1974 respectively). Progress is made, but the stunning impact sought has not been made. Consider then the records of the one truly interactively computerized instructional center in the country today, namely, at Dartmouth.* On a campus of 3500 students in fiscal year 1970-71, 2587 undergraduate students, 448 graduate students, and 346 faculty members used the computer. Usage was spread campus wide with the following usage recorded in the half-year February to June 1970:

Area	% of Total Use
Sciences	47%
Social Science	43%
Humanities	10%

In February of 1971, the number of interactive sessions per day was 2449, from a total of about 150 teletypes on campus. In brief, well over half the students at Dartmouth use the computer

---

* Weingarten, Nielsen, Whiteley, Weeg, A Study of Regional Computer Networks, University of Iowa, 1972.

(indeed, except for repeat runs it could appear to be 90%). A large number of faculty members have devoted extensive amounts of time in producing radically new teaching formats, and we believe the result has been shown in a superior product.

At the University of Iowa, about 15% of usage had been for instruction, 85% for all other purposes. It is estimated that no more than 15% of all classes made use of computing. We too are attempting to promote instructional computing use, but we are convinced that we were hampered both by the saturated batch systems we use and the simple failure to provide interaction.

We maintain that given massive numbers of terminals, adequate computer backup, a simple computer language, and an intense educational activity among our faculty members, with guarantees that computing would remain available in adequate supply, that the faculty would re-think its courses so as to incorporate computing at logical junctures. We believe that in the span of a few years, we would revolutionize education on our campus.

5.2 Services Needed We should proceed to provide adequate instructional computing capacity; we must explore the promise of CAI; we must continue to meet the research needs of our faculty and students; and we must see to the continued soundness of our computer center staff in light of these developments.

Since the university has as its principle objective the instruction of its students, we have proposed to move on undergraduate instructional computing first. To do so, we must provide a vast number of interactive terminals. The exact number to be so provided is amenable to simple arithmetic. From the aforementioned Weingarten et. al. report on networks, it appears that six terminals are needed per 250 students who receive one regular computer assignment per week. If our objective therefore is to have 50% of our 20,000 students using the computer for instruction in an intensive fashion, we need 40 times 6 or 240 terminals, minimum. If those students have several computer oriented courses instead of just one, the number of terminals must increase by some multiplicative factor. Hence, we believe that there is a need for terminals ranging from dozens now to many hundreds in a few years. It is completely impossible to meet the need of such numbers of users of interactive systems through our IBM 360/65, or to our knowledge, by any other single computer available today. We can always add one or two more terminals, but we certainly can not add the hundreds which are needed. The experience at Dartmouth shows that 95% of the need for interactive computing is met by an extended BASIC language, which is chosen by most even when more sophisticated languages are available. We therefore have proposed the establishment of satellite centers throughout the university, composed of a mini-computer which will drive 32 teletype-like terminals in BASIC, and also allow access to central files and the batch stream of the UCC central computer. Card readers and line printers might be added to some of the satellite centers at low cost to provide remote batch facilities.

We have also proposed that the use of the BASIC system so established be made FREE TO ALL STUDENTS, FACULTY, AND STAFF OF THE UNIVERSITY. Since the systems can be purchased at relatively low cost, only maintenance and minor operating costs would ensue after installation, and the university can and should assume this cost.

To provide for the exploration of CAI on this campus, we have established a center of expertise within the newly completed Lindquist Center (our new building) and have appointed a new Director of Computer Assisted Instruction Research and Development. Under his direction, several developments will be pursued simultaneously. These will include increased use of COURSEWRITER III on the UCC computer, and installing a mini-computer dedicated to CAI with up to 32 terminals. This equipment is intended to promote research and development only. If and when certain areas determine that CAI as a production tool is important, additional sources of CAI power will have to be arranged.

So as to update our batch processing capacity at least to the level where the community can once again consider expanding its activity, it is proposed to upgrade our 360/65 to something like a 370/168. The staff also will be enhanced, particularly at the level of systems programming, applications programming, instructional specialists, and communications experts.

Finally, if we are going to turn education around, we must provide educational activities for the faculty (workshops, travel to other centers of excellence, released time, fellowships and the like).

5.3 Factors of the Plan Our plan for this interactive computer based revolution involves carefully coordinated hardware selection, software implemention, professional staff dedication, and a major effort to train faculty members in the new technology, all preceded by a carefully planned prototype experiment. As faculty training is the most critical part of the whole enterprise, we discuss it first.

5.3.1. Faculty Training - We seek ultimately to have an aroused faculty teach in a way that they presently do not even suspect. We maintain that this job will be impeded by the lack of adequate course materials for faculty members to resort to. Hence we need a two pronged attack: first, it is necessary to provide general training in depth of what interactive computing is and the promise it holds in individual areas: second, faculty members must be given time to create the new material which must accompany this revolution. The first chore is reasonably easy in concept - we can teach faculty members what they need to know about computing. The second chore is hard and will take time. As a case in point the Iowa Regional Computer Center carried out a two year NSF-supported project to revise the undergraduate mathematics and biology curricula so as to incorporate computing in beneficial ways. At the end of two years half the material was indeed valuable, half was much less so. We have also sponsored summer faculty fellowships to do the same thing, but most of the proposed projects remain unfinished after two years.

Hence, we see a need for general teaching sessions and a large number of faculty fellowships to begin the work of revision. We do not stand alone in this field - the CONDUIT (Computers at Oregon State, North Carolina, Dartmouth, and the Universities of Iowa and Texas) consortium has been established to discover good curriculum material and to promote it at diverse centers.

5.3.2. Computer Center Staff - To serve these new needs, the staff of the Computer Center must be augmented by learning specialists, programmers, and analysts. These will spearhead the development.

5.3.3. Hardware - We intend to introduce a large number of mini-computers on campus, while continuing to provide for upgraded batch capacity.

All mini-computers will be located at the Computer Center proper, with terminals distributed to appropriate locations via telephone lines or hardwired cables. The maintenance of hardware and software will be expedited in this fashion, and the linking of these mini-systems to the central computer will be more efficient.

5.4 The Prototype System We restricted mainframe selection to two mini-computers at first, though as technology advances this decision will be reviewed. We specifically considered the PDP 11/40 and the Hewlett Packard 2000F. The prototype system is a

HP2000F, placed in a controlled environment, to determine the effect adequate terminal facilities would have on faculty and students. Hence, sixteen terminals have been placed in the College of Business Administration, with use restricted as much as possible to that college alone; with use primarily directed toward instructional purposes; and with use being at no cost. The fairly complete program packages of the business schools at Stanford and the University of Chicago have affected this choice. The ratio of students (less than 1,000) to terminals in that situation seems to be adequate. Admittedly, this may be an oversupply initially, but it is important to have enough to encourage long range development by faculty.

The remaining sixteen terminals have been placed in two colleges, eight being placed in the College of Education and the other eight in the College of Liberal Arts. These two colleges were chosen not because eight terminals can serve such large colleges, but rather for maximum impact. Through intensive efforts of several individuals, selected faculty members will be encouraged to change their teaching techniques so as to incorporate this new tool.

In Liberal Arts, the terminals are dedicated to development in five departments (selected because we believe there is a group of faculty members ready to work on this innovation), namely: Geography, Sociology, Political Science, Psychology, and Linguistics.

Before installation of the prototype system, a concerted effort to prepare for the system was made, including site planning, selection of terminals, training of faculty members in computer use, identification of trial units for early development, and preparation of the UCC staff. To accomplish all this, planning committees were created in Business Administration, Education, and Liberal Arts. These committees work with the Computer Center, and are coordinated by the office of the Vice President for Educational Development and Research with the assistance of the (campus-wide) Computer Based Education Committee.

Each committee is responsible for site planning, training and for presenting regular progress reports on the affect the system has on education. The Computer Center maintains detailed use records.

The terminals selected for the prototype system were as follows:

8 - Model 38 Teletypes without paper tape (KSR)
8 - Model 38 Teletypes with paper tape (ASR)
16 - Model IIAE Beehive Cathode Ray Tube (CRT) terminals

The teletypes were selected because they have both upper and lower case letters in their standard character set, and because they were the cheapest keyboard-printer terminal available of this type. The Beehive terminals were selected after examining a wide variety of CRT terminals then available, which had upper and lower case characters, variable transmission rates, and the capability of accepting an optional printer through a standard interface, as well as being reasonably priced.

Each college's committee decided how best to install the terminals for their own use. In the College of Business Administration, which received half of each type of terminal, it was decided to remodel and connect two adjoining rooms, which house all 16 terminals. The College of Education made a similar decision, locating six of their eight terminals in a room adjoining their Curriculum Materials Center, but reserving two terminals (one teletype and one CRT) for somewhat portable use, since their faculty is widely spread over a number of buildings on the campus.

Since the committee from Liberal Arts had the problem of serving five departments, each in a different building and separated by as much as six blocks, they decided to use a combination of centralization and decentralization. A (former) classroom adjoining the Laboratory for Political Research, which is affiliated with the Political Science Department, houses a remote batch (reader-printer) terminal to the Computer Center and supporting equipment, and is located in a building which is central among those involved, has been assigned as a terminal room and houses four of their terminals. The remaining four terminals are located at the respective departments of Geography, Linguistics, Psychology, and Sociology for developmental use by faculty and graduate assistants. This dispersion would be undesirable for access by large numbers of students, but will suffice until such time as "production" programs are in use and (hopefully) more terminals can be provided. All five departments will, in the meantime, send students to the Political Science facility.

5.5 Cost Considerations The system selected initially, the Hewlett-Packard 2000F Timesharing System, consists of the following:

Main Processor, with 32,000 words of memory
Input-Output Processor, with 8,000 words of memory
Paging Drum
Tape Drive (9 track, 800 BPI)
Disk Drive (23.5 million words of storage)
Multiplexors for 32 terminal ports
Line Modems (Com-Data), up to 300 baud rate
Terminals (Beehive and Teletype), as mentioned above

The system priced out at approximately $215,000, complete. Through price reductions and the recent discovery that the system will run essentially the same even though the (expensive) paging drum is removed, the system (referred to as the HP 2000F, option 200) can now be installed for approximately $165,000 (complete), or about $5,150 per terminal. As the cheaper CRT terminals now being introduced become reliable, we anticipate that the per-terminal cost will drop to no more than $4,500, and might soon approach $4,000.

We acknowledge that a comparable, or (temporarily) perhaps slightly better, cost per terminal could be obtained through installation of a very large processor, capable of supporting hundreds of terminals in and of itself. However, we quickly abandoned this approach, for four reasons:

1. Mini-systems require very little regular attention from operators or systems programmers, thereby reducing ongoing expenses,
2. Reliability problems are minimal, both because minicomputers are usually more reliable than large computers, and because problems on a single system only affect 32 terminals (maximum) at a time,
3. The modularity of this approach, requiring only several hundreds of thousands of dollars at a time instead of several millions all at once, is much more palatable during times of tight budgets, and
4. By pacing the installation of mini-systems over a number of years, we can take advantage of the rapidly-evolving advancements in the technologies of minicomputers, terminals, and communications, thereby conserving funds and at the same time keeping up with the latest capabilities available.

5.6 Enhancements to the Mini-Center As we recognize that the mini-systems described above cannot serve all needs, or even just the needs met in the typical undergraduate course, we have also planned a number of enhancements to the centers, to be achieved by revision of the standard operating software supplied with the system. Over the 1973-75 academic years, the following changes are being made to that system:

1. Incorporation of modules which will allow a card reader and line printer, attached to the HP 2000F and

operating simultaneously will the timesharing system, to communicate with our IBM 360/65 by standard HASP protocol. This will therefore provide remote batch capability at each center where such readers and printers are installed, referred to as Remote Batch Entry (RBE).

2. Incorporation of modules which will allow access to the batch system on the IBM 360/65 directly from the keyboard terminals on the HP 2000F. This is referred to as a Remote Job Entry (RJE) facility,
3. Incorporation of modules which will allow communication between the IBM 360/65 and the HP 2000F so that files built on the HP 2000F can be stored on the large disks of the IBM 360/65, and retrieved for later use on the HP 2000F, referred to as Remote File Transfer (RFT), and
4. Incorporation of modules which will allow terminals tied to the HP 2000F to be linked directly with the IBM 360/65 for the purpose of accessing the timesharing systems available on the large computer, through using the HP 2000F as a Remote Terminal Concentrator (RTC).

These enhancements do have a price tag attached, since hardware must be added to the system to handle the additional load. These items are an additional 16K of storage in the I/O processor, Direct memory Access (DMA), Writable Control Store

(WCS) for microprogramed instructions, and a synchronous interface. The cost of these items is about $10,000. In addition, if a card reader and line printer are to be used, they must also be purchased, with the cost depending on the units selected.

5.7 Staff Commitments Obviously, the efforts required to accomplish these plans require significant commitment of the staff of the Computer Center, as well as affected departments and learning specialists throughout the campus. As it is the practice on our campus to encourage joint-appointments of learning specialists between the College of Education and the various other disciplines, a significant body of expertise already existed. Many of these faculty members have volunteered all the time they can spare, and a large grant application is now pending to support faculty in a truly dedicated fashion over several years.

The Computer Center's contribution to this effort is very large, and consists of two main groups of staff. The first is the "interface" team, which meets with their counterpart committees in each of the affected colleges. This team is composed of eight staff members, distributed as follows (the staff position of each at the Computer Center is given in parenthesis):

Team Leader (Assistant Director)
  System Development (Manager of Development)
  Terminals & Communications (Manager of Terms. & Commun.)
  Remote Site Physical Planning (Manager of Networks)
  Central Site Physical Planning (Operations Manager)
  Faculty Training (Manager of Academic Services,
    Manager of Curriculum Development,
    and Computer-Based-Education Specialist)

This does not of course, represent full-time commitments, but it does involve significant amounts of time for most of the first-line managers on the staff.

The second group of Computer Center staff is the "development" team, which is responsible for implementing the enhancements mentioned previously. This group is led by the Development Manager, and consists of five senior programmers selected from the systems and applications groups. They are working in two teams, developing phases which can be overlapped. Again, this is not a full-time commitment for all involved, although 60 man-months of effort are anticipated during the term of the development.

6. PROGRESS WITH THE PLAN

6.1 An Early Timetable No one is certain of the best rate for the overall plan to proceed. However, about two years ago a special ad-hoc committee, formed by the Vice President for Educational Development and Research, recommended a schedule intended to provide for all of the research and instructional computing needs of the campus over a two and one-half year period. The plan is presented below.

Equipment	Location	Time
Mini-Center; 32 Terminals	Bus. Admin. (16) Education (8) Liberal Arts (8)	July, 1973
2 PLATO Terminals	C.A.I. Laboratory	July, 1973
Mini-Center; 32 Terminals	Liberal Arts	July, 1973
Mini-Center; 32 terminals	Engineering (16) Dentistry (16)	Jan., 1974
Mini-Center; 32 Terminals Reader & Printer	Medicine	Jan., 1974
IBM 370/168	Computer Center	July, 1974
Reader & Printer	Liberal Arts	July, 1974
Mini-Center; 32 Terminals	Law (8) Nursing (8) Education (16)	Jan., 1975
8 PLATO Terminals	C.A.I. Laboratory	Jan., 1975
Mini-Center; 32 Terminals	C.A.I. Laboratory	Jan., 1975
Mini-Center; 64 Terminals	Liberal Arts	July, 1975
Mini-Center; 32 Terminals	Education (16) Computer Center (16)	Jan., 1976

The total cost of this plan, including equipment, faculty, and Computer Center staff, would be over $5,000,000; about 60% of that cost would attribute to the installation of the new large central computer.

6.2 Acquistions to Date As of this writing, four minisystems have been obtained. One is still in use for development, one is installed in the CAI Lab, and the other two are in use in Business, Education, Liberal Arts, Dentistry, and Engineering.

One PLATO terminal has been installed, but we probably will not obtain others. The projected 370/168 is still in the future.

6.3 Software Development to Date We are currently completing the development of all of the projected capabilities except remote terminal concentration, which will be handled separately because it requires either adaptation in the host (IBM) computer or the insertion of a programmable front end between the host and remote systems. We are behind schedule, but for good reasons. About eight months ago we learned from sources within Hewlett Packard that a new operating system was being developed, and we were faced with the choice of continuing our development for the existing system, or coordinating our efforts with the internal development in HP.

We chose the latter course, and this has slowed the development; however, the result will be that both we and HP will be able to take full advantage of each other's enhancements. Accordingly, when the new version of the HP software is released, it will contain all of our additions to the I/O processor of the HP2000F, (except remote terminal concentration) as well as all of the enhancements which HP has added to the system processor in this two-processor configuration. This not only will give us more capabilities than we originally envisioned, but the interrelationships which have taken place through this joint-development period are highly valued by our staff, and, we hope, by the corresponding staff of Hewlett Packard as well.

By way of a timetable, we expect all of the resulting software to be finished and tested, and therefore ready for production, by late spring. The remote terminal concentration capability will be developed during the coming spring and summer, and we hope to implement it in the fall of 1975.

6.4 Curricular Developments While we have only one full year of experience with the mini-systems, and our enhancements are not in production, significant effects are already being noticed in the field. Our College of Business Administration has 800 declared majors (Juniors, Seniors, and Graduate Students), but also has about 700 pre-majors (Freshman and Sophomore) who take a specially designed Mathematics-Statistics course sequence for a year. Virtually all of these students are now using the 16 terminals located in the Business College, which has placed a strain on the accessibility of these terminals, but this is a very positive sign and creates the kind of problem we enjoy solving. While use is not yet so heavy in the other areas supplied with terminals (Education, Engineering, and Liberal Arts), it is steadily increasing, and a great many courses are under revision to incorporate all types of interactive computing.

In addition, a number of departments not now served by terminals (except through a cluster in the Computer Center) are developing new or revised courses in anticipation of the availability of terminals on future systems. These include programs in Health and Hospital Administration, Law, Nursing, Drama, Mathematics, Statistics, Physics, Classics (Latin and

Greek), and Medicine. While most of these are at present restricted to one or two courses in the department's list of offerings, all bode for steadily increasing usage in the years ahead.

6.5 Research Applicability Our efforts with these systems have been aimed almost exclusively toward instructional applications. Indeed, we have actively discouraged the use of these systems for research purposes, other than research about the effectiveness of the programs themselves. This is because we are so limited in the number of terminals we have to offer, and wish to avoid having an instructional application wither for lack of terminal access.

This will not always be the case; once sufficient terminals are available, and particularly with the enhancements to the systems which will then be available, we will have no objections to the full use of the terminals for both research and instruction. Such use will come easily and naturally, as already has been discovered by some members of the faculty who have had to be cautioned against creating impediments to the ongoing curricular developments.

6.6 A Somewhat Different Application While this general development is intended to support instruction and has some natural extensions into research, we do have one example of a completely different approach to its use. Our College of Dentistry has for several years been investigating ways in which they could record and control all of the data they collect on the students and patients in their teaching clinics. The same experiences that provide professional growth for the students are also treatments for patients, and the relevant data therfore forms a naturally integrated data base for administration of the College.

When we began our development, the College closely evaluated both the existing and future capabilities of the proposed system, and concluded that this arrangement could serve their needs quite well. Accordingly, they purchased a mini-system with various funds allocated for equipping their new building, and began to develop the data base and programs to access it. Their system was first placed into operation this past September, and is so far performing quite well.

In concept, the system is quite simple. Basic information about patients is entered into the mini-system from registration forms used in the initial contact with patients. From that point on, an electronic "ledger card" in the file accumulates all other information about the patient, including the nine most current visits (any others "fall off" the system into a manual file, which is exceptional), all procedures performed on the patient, the student or students who have performed these procedures, the date and time of the next visit, and the debit and credit record of each patient from the cashier's office. The necessary updates are made by the clinic clerks from small forms filled out in the clinics as patients and students meet.

The necessary backup for such a system is maintained in two ways. First, as each transaction is entered through one of the 14 CRT's at the clinic stations and/or the Cashier's Office, it is also recorded on a disk file and then dumped to a magnetic tape cassette on a master terminal. The tape cassette arrangement is temporary, until the capability of directly accessing the tape drive on the time-share system becomes available (anticipated in about 6 months). Second, maintenance programs which carefully check the files are run daily, in case unanticipated problems of any sort should get past built-in checks during the day. To date, only a very few minor problems have been found, and these have been easily corrected.

The above procedures explain nothing about student data, and what is done with it. The same update slips which contain patient information also include student ID's, records of procedures performed, and any comments by the instructor. This data is temporarily stored in the form of print files on the HP timeshare system, and then analyzed on the IBM host system. Eventually, the system will also participate directly in the insturctional process through such programs as case-study simulations, which are now under development.

This system, dubbed the POLIDENT (Patient On Line Information DENTal) system by its creators (with no pun or commercial message intended), seems to us to provide substantial evidence that the flexibility we seek is inherent in our development, and is the therefore quite encouraging.

7. WHAT DOES THE FUTURE HOLD?

7.1 Wiring the Campus As I stated earlier, we hope to provide easy access to terminals to every student and faculty member on the campus. Since we have no rich uncle, we must gradually expand our hardware as finances permit. Currently included in our legislative askings are enough funds to provide one new mini-system each year, and we hope to continue doing this indefinitely, or until it can be determined that we have a sufficient number of terminals. But we also have a fully loaded (and owned) host system, and no funds to replace it. We are in the process of developing a plan for financing a new central system, with at least the capabilities of an IBM 370/168, for installation in 1977. While we'll be hard pressed to get by until then, if we are successful we will at least have removed the physical limitation to continued development.

7.2 Evaluating Success This is not a short-term plan, nor can we determine success on a short-term basis. We have been successful in beginning, and have acheived the encouragement not only of the faculty but of the central administration of the University. The true test must await the reactions of the faculty and student body and, ultimately, the extent to which the curriculum is materially affected. Only if measureable improvement in learning occurs will we be satisfied that we have reached our goal.

THE DEVELOPMENT OF THE MASSACHUSETTS STATE COLLEGE COMPUTER NETWORK

Leo Roomets
Assistant Director
Computer Services
State College Computer Network
Massachusetts

Computer Networking in Massachusetts

Leo Roomets

INTRODUCTION

The Massachusetts State College Computer Network was established in October of 1973 to provide a computing and data processing resource for the eleven state colleges of Massachusetts. The network itself consists of a Control Data Cyber 72 computer located in Boston which supports both remote batch and interactive terminals located on each campus tied in via a state Telpak network. Patterned after the classic star configuration, the centralized network currently bridges the developmental gap between what was once a cluster of small independent computer operations and what will eventually be a hierarchical computing network.

The distinguishing feature of the network, however, is the variety of functions which it is designed to support and the services which it offers. Breaking traditional academic-administrative detente, the system provides a single resource for all instructional, administrative, research and public service computing requirements of the colleges. Additionally, the network center supports a systems and programming staff charged with the responsibility of developing and implementing an integrated information system designed around linked data base structures and intended for use in an on-line, real-time transaction-oriented environment.

In order to maintain fairly a multi-institutional user-oriented service, the center has been established as an organizational entity separate from the colleges, and its management reports to the Board of Trustees of the State College System through the provost of the system.

Computing is a relatively new addition to a system which has a history that dates back to 1839. Although this factor has been the source of innumerable headaches and general technology shock, it has also been the singlemost significant factor which has allowed the colleges to leap-frog two decades of technological growth and development in both hardware and software concepts in less than three years.

In 1965, the state colleges acquired unit record and tabulating equipment to support the processing of student and financial data. As the colleges increased their enrollment, small computers were leased by several colleges to meet the growing data processing requirements. Although usage was still primarily administrative, a few colleges acquired minicomputers to be used in teaching physics, chemistry, and mathematics. The year 1965 also marked the year when the state colleges were organized under a single Board of Trustees with governing responsibility. This change presented individual colleges with the added task of responding to many new requests for information concerning all the activities of the institutions, requests which the traditional information collection and processing procedures were not designed to handle effectively.

The late 60's also marked a period of change in the public attitude toward traditional unquestioned state support of public higher education. The Board of Trustees of the State College System felt the pressure for accountability, for curricular relevance, for effective resource management, and for justifiable expenditures of public money. In responding to increasing criticism and closer scrutiny of the capital and operating budgets of the colleges, trustees and presidents alike realized the need for more meaningful and accurate information to defend their requests and their programs.

Furthermore, computers in general were penetrating into all facets and levels of business, education, government, and industry. As a result, both faculty and students were putting pressure on the colleges from within for more sophisticated and more accessible computing capability; employers as well as graduate schools were looking for students with computer skills.

It was in the context of this environment that the use of computers to support academic and administrative requirements began getting special attention in 1971. A Staff Assistant for Management Information Services was hired by the Office of the Board of Trustees of State Colleges to analyze and document the management information systems and the instructional support systems required by the State College System, and to recommend a plan of action which would result in the development of an effective and economical computing resource. This feasibility study indicated to the trustees that: 1) the management information needed by the trustees, presidents and college administrators could be supplied only by information systems and procedures which were considerably more advanced and sophisticated than those which were currently in use; 2) the faculty and students could benefit significantly through the expanded use of the computer in a variety of disciplines; and 3) the proliferation of independent computing solutions (service bureaus, local hardware, consortia) could not provide an effective, cost-justifiable capability to meet short-term or long-term needs. The major recommendation of the feasibility study was to develop a computing master plan which would contain detailed information system designs, guidelines for the development of instructional computing, an implementation plan and timetable, hardware and software specifications, and personnel and budget requirements for the next five years. Because the final plan had to be acceptable to all members of the college community before it would be approved, the planning process had to allow for considerable user input and evaluation to insure that the designs and recommendations were consistent with college requirements.

Master Plan for Computing

The details of the planning project were worked out, and in May, 1972 the computer services staff was increased to four analysts. In June, they began the analysis and design work assisted by Systems Research Group, Inc. After eight months of interviews, meetings with college and system-wide committees, and presentations to administrators, presidents, and trustees. the plan was completed. The trustees voted approval of the Master Plan for Academic and Administrative Systems in February of 1973. The Plan, documented in five volumes and over 2,000 pages, was perhaps the most comprehensive set of designs and specifications ever developed for a group of colleges. Incorporating the experiences of other systems of higher education like the California State Colleges and Universities, the State University of New York, New Jersey Educational Information Services, Inc., and others, the Massachusetts plan included state-of-the-art concepts in systems design, teleprocessing, and network organization, in addition to management system concepts developed by the National Center for Higher Education Management Systems.

The Master Plan contained:

Volume I, Summary and Recommendations, consists of a summary of the project recommendations to the Board of Trustees and a plan for implementation.

Volume II, Analysis of Current Operations, consists of a detailed analysis of the academic and administrative data processing activities at each of the colleges.

Volume III, Academic Computing, consists of a discussion of the current and future use of the computer in instruction, research and public service. It also includes specifications for a Curriculum Evaluation System and a Research Management System.

Volume IV, Administrative System Specifications, consists of detailed specifications for the State College information system. General description, flowcharts, process descriptions, definitions of inputs, files and outputs are provided for the following subsystems:

- Planning and Budgeting
  Program Planning
  Multi-Year Budget Planning
  Simulation Modelling
  Facilities Planning

- Statistical Reporting Systems
  Curriculum
  Finance
  Personnel
  Physical Facilities
  Students

- Student Systems
  Administration
  Financial Aid
  Residence
  Registration
  Grade Reporting/Transcripts
  Counselling
  Alumni Reports
  Attrition Analysis
  Student Records

- Financial Systems
  - Accounting
  - Budgetary Control
  - General Ledger
  - Payroll
  - Purchasing/Inventory Control
  - Accounts Payable
  - Accounts Receivable/Student Billing
  - Cost Analysis

- Personnel Systems
  - Personnel Records
  - Assignments
  - Activity Analysis
  - Collective Bargaining/Contract Management
  - Personnel Evaluation

- Physical Facilities
  - Facilities Inventory
  - Facilities Utilization
  - Facilities Management & Planning
  - Maintenance
  - Capital Budgeting
  - Building Project Control

- Curriculum Systems
  - Program Analysis
  - Program Evaluation
  - Course Catalog Curriculum
  - Curriculum Planning

- Institutional Support Systems
  - Library acquisition, cataloging, and circulation control

Volume V, Data Element Dictionary, consists of a cross-referenced index of all data element codes and names used in the Administrative Systems Specifications.

Network Development

Upon acceptance of the Master Plan, the Computer Services Staff began all the activities which were required to implement the plan's recommendations: building of a computer network; hiring and training of personnel; and development of systems to be running by the Fall of 1973.

The central site was prepared and the CDC Cyber 72 installed on October 1973 as scheduled. By November, the eleven state colleges were all accessing the system through the time-sharing and remote batch systems and processing their first jobs over the Massachusetts State College Computer Network. After the first two months, during which the operating system (KRONOS 2.1), the hardware, the telecommunications network, and the personnel went through an initial shakedown period, the network began offering reliable services to the administration, faculty, and students at the colleges. Usage of the network grew from roughly 2500 jobs per week to nearly 20,000 jobs per week by the end of the Spring 1973 semester.

One year has been allowed for the conversion of college data processing from dependence on local hardware or service bureaus to complete reliance on network resources. By June, 1974, many of the colleges have already completed the conversion of all software, and have released equipment which is no longer required for network processing. Other colleges are still running dual systems, but expect to be totally on the network by the Fall of 1974.

To assist in the conversion, the network center has developed an administrative software library and clearinghouse for interim systems. Programs written at the Massachusetts state colleges or acquired from external sources such as CAUSE, or other colleges and universities are available for use by any of the user institutions. Internal consulting services, user training, and programming assistance have also been provided to facilitate the changeover to the network.

Network Services and Activities

The computer center is organized and equipped to provide a full range of services to the network users. The CDC Cyber 72 computer system supports a variety of programming languages that will satisfy nearly any user's needs. The time-sharing sub-system allows interactive programming in BASIC, FORTRAN, and APL; the batch sub-system allows local or remote processing in COBOL, FORTRAN, COMPASS assembly language, ALGOL, SIMSCRIPT, SIMULA, and PERT-TIME, all of which may be used through interactive terminals as well. In addition to these languages, the system also provides users with a software library of prepared programs which can be used to perform a number of functions. Programs are available to perform statistical analyses (e.g., SPSS - Statistical Package for the Social Sciences, correlations, analyses of variance, regressions, etc.), general mathematical problems (e.g., linear programming, matrix algebra, calculus, etc.), graphic plotting (e.g., coordinate plotting, graphs, geographic mapping, etc.), financial analysis (e.g., costing, forecasting, interest compounding, etc.), engineering analysis (e.g., steel beam design, circuit analysis, etc.), operations research (e.g., GPSS - General Purpose System Simulator), and many other computational functions. Current plans call for the addition of programs from the Dartmouth Time Sharing System Library, from the Harvard Business School Library, and from other colleges and universities. A tutorial language, PLANIT, will also be available to allow faculty to write programs to be used by students in a computer-assisted instruction (CAI) environment. Administrative packages developed at the state colleges or at other institutions can be used by anyone on the network interested in areas such as student records, billing, grade reporting, etc.

In the event that existing packages are not consistent with a user's requirements, the computer center has a highly-skilled professional staff of analysts and programmers who can assist a user in developing his own programs or they can design and program systems for users who do not have their own technical staff. If necessary, the computer center can also assume the responsibility for the operation of systems which have been developed and are in the production stage.

Additional services which are available to the network user community include training in computer skills; design, implementation and maintenance of telecommunications interfaces to the computer system; and purchase or lease of peripheral equipment through bulk contracts with vendors.

All of the above services are provided to the eleven state colleges for no charge. A relatively sophisticated resource accounting and charging system has been developed based on a simple cost-recovery algorithm which gives each state college user a complete breakdown of the system resources he has used and their cost. Although no payment is transacted, the accounting information gives users a good feeling for how much or how little his job would have cost, thereby avoiding as much as possible the development of inefficient usage of what could be perceived as a totally "free" and unlimited resource.

Many state agencies and other educational institutions have been attracted by the software and services available through the network and have entered into user agreements with the network. External users have been permitted so long as sufficient resources have been available (or added as part of the user agreement) to maintain a consistently high level of service to the primary user community, the state colleges. Adding additional users has promoted the more efficient use of the system resources (e.g., the network now operates 24 hours/day, seven days each week except during periods of preventative maintenance or major systems modifications), and has, in some instances, justified the addition of equipment or programs which are automatically available to all users. Carried out within reasonable constraints, such a policy results in increased benefits to all users while decreasing the actual cost to any single user or user group. All external users, with the exception of those in the public service category, pay for the resources and services which they need. Some of the current and pending projects or services involving external users include:

- Board of Higher Education Statewide Scholarship Processing System - a complete service from design and programming through operation and maintenance, and which involves both local batch and remote timesharing capability.

- Northern Essex Community College - academic timesharing service for four interactive terminals.

- Massasoit Community College - academic timesharing service for two interactive terminals.

- University of Massachusetts - timesharing and remote batch service to support conversion of UMASS to new computer system compatible with MSCCN.

- Department of Correction - local batch processing services to support the analysis of data using the Statistical Package for the Social Sciences (SPSS).

- Division of Water Pollution Control - remote batch processing services to support the simulation and analysis of water pollution control measures.

- Office of the Secretary of Education - timesharing services to support the development of analytical programs.

- Massachusetts Advisory Council on Education - local batch and graphics services to support the analysis of statewide regionalization of school districts.

- Legislative Committee on Unequal Educational Opportunity - local batch and graphics services to support the analysis of educational opportunity across the Commonwealth.

Other community colleges and state agencies, as well as a number of organizations or municipal agencies located near the colleges, have indicated a strong interest in using the computer network. The network has the growth capacity to handle eventually up to more than one thousand interactive time-sharing terminals and over one hundred remote batch terminals. And because of the increasing economics of scale afforded by large computer systems, the network will be able to support the growing computing activities at the colleges as well as additional external users at a marginal cost that would be a fraction of the cost of developing additional independent computing resources.

PLANNING, INSTALLING, AND USING REMOTE BATCH TERMINALS

G. E. Bird
Manager
Data Services
Union Carbide Corporation

PLANNING, INSTALLING AND USING REMOTE BATCH TERMINALS

G. E. Bird, Data Services Manager

Union Carbide Corporation

Tarrytown Technical Center, Tarrytown, New York

Trying to predict the future of telecommunications is an impossible task. However, as Data Processing Managers, we are asked to make such decisions in spite of major changes in regulatory policies and sweeping technological advances. It is my belief that the best a manager comtemplating the use of remote batch terminals can do today, is to understand the "environment." This understanding should be considered or remembered regardless of the phase of planning, installation or use. In order of priority they are the psychology of the user, changes in technology, systems growth, and changes in communications regulatory policies.

1. PLANNING

Needless to say, planning is the single most important function in the process of remote batch installation and use. The planning function should involve top management of the corporation, location management, and most important, the remote user himself. Involvement of these people will greatly enhance the project's success and avoid the adoption of parochial attitudes toward the effort. The remote user should become the single most important input into the planning function. Only he can identify his computing requirements and what must be involved for him to do his job. The first effort in planning should be a field survey to

PLANNING cont'd

to gather data regarding volumes, schedules, programming needs, physical site preparations, operator training, etc., etc., and to expose the user to his new position. Secondly, data should be collected for specific hardware, software, communications requirements, and data collection techniques that meet each user's requirements. If a feasibility study is done or if it is determined that the remote batch terminal will be installed, the plan should evolve to a work plan that includes each major sequence of activity and identified responsibility for each. With costs being a major factor for remote batch installation, long-range activities and goals should be considered to avoid switching the terminal gear and the trauma associated with it. The plan should cover a two-year time frame with expected activities identified and understood by all parties.

2. VOLUME AND DISTRIBUTION

Geographic distribution of sites and data volumes generated and received by each remote location will influence the selection of remote equipment and possibly the central Site computer. Communications volumes may be defined in many ways: Words per minute; cards per minute; bits per second; lines or characters per second. Data volumes should be expressed in a common language such as bits per second to avoid any misinterpretations of the actual data volumes involved in servicing a remote location.

VOLUMES AND DISTRIBUTION cont'd

Data volumes and remote locations are plotted by geographical distribution. The systems designer can then analyze the communications problem, determine the proper mix of facilities, and utilize equipment and communications techniques that will handle the data volumes economically. Areas of concentration, circuit layout, and communications methods can be determined. Very often the resultant system will use a variety of communications techniques and facilities to handle the data volume.

3. SPEED AND COST

While it is possible to transmit data at very fast speeds, the cost of line facilities and terminal equipment has been a limiting factor for commercial applications.

The characteristics of the remote equipment will, to a large degree, establish the type of communications facilities used. In each speed range, both equipment and facilities have different costs and performance traits. The ultimate network may use a combination of different equipment and facilities to meet the requirements of volume and response time. Speed alone does not reflect efficient system design. It is essential to face the economic realities that higher speed remote equipment requires higher speed communications facilities and produces inherently higher costs. The aura of sophisticated circuits and remote equipment should not distract from achieving a balance between volume and cost.

4. INSTALLING

Physical installation of the terminal becomes a very easy job if the planning has been done properly. I have found that the worst part of a new installation is avoiding the finger pointing that happens in a multi-vendor situation. In most cases, where the telephone line, modem, terminal and host processor are different vendors, each will be responsible for only his own equipment. This means connecting power and doing surface diagnostics. It becomes the user's responsibility to make all interface connections and to make them work together. To avoid the finger pointing of vendor to vendor, we have purchased a relatively inexpensive test set called a "Data Interface Test Set." This device has LED indicators for each RS 232 lead, and readily pinpoints each lead status. It has been a tremendous help to me in avoiding calls to the wrong vendor, and vendor finger pointing.

5. USING

Using a computer that is physically removed from the user entails great responsibility on the part of the service organization. The operational success of the terminal depends upon how well the central site communicates to the end user. Users must have access to people who can help them on a one-to-one basis, and should be kept advised of whom to contact.

Do not forget the fact that at a remote site there is not a pool of trained personnel versed in the use of the hardware. Moreover, on-site system experience doesn't necessarily carry over to remote operations.

USING cont'd

To minimize these problems, design your system and operational techniques so that the key punch operator can follow instructions and dual as the terminal operator. Aspects of the system, languages, or procedures that users are likely to encounter should be explained by examples and demonstrated periodically.

Because of technological advances of the terminal manufacturers, we have today an ever increasing requirement for communications between a wide variety of terminal types. The catalyst for this requirement has been the development of remote terminals and special purpose terminals, each providing unique and desirable features. At the time of selection, it is important that all of the terminal factors be considered. The requirements of the data types to be handled will decide the type of terminal needed; i.e., hard copy during input, printer and card or tape buffering may be required for printer output. Also, the characteristics of this data will determine the best mode of operation. A full-duplex line is only slightly more expensive than a half-duplex line, but moving traffic in both directions at the same time may not be possible in many applications. The content of the data to be transmitted will determine the type of error checking and correction routines.

As new services, facilities, tariffs, etc., evolve, the use of batch terminals will be an ever changing ocean and will require your continuous attention.

6. SUMMARY

Under a well designed system, it is virtually impossible to segregate the data processing area from the communications area. The communications system to operate at peak efficiency, will tend to lose its identity as a separate unit and will become, instead, an integral part of the input and output of the data processing system itself. A thorough understanding of the capabilities of communications facilities, remote equipment, and how a system can use them, will result in superior performance and economics.

/jc

WHEN ARE MICROFILM SYSTEMS BETTER THAN ON-LINE SYSTEMS: CAN COM COMPETE?

Alex Wenger
Director
Information Resources
Temple University

WHEN ARE MICROFILM SYSTEMS BETTER THAN ON-LINE SYSTEMS: CAN COM COMPETE?

Alex Wenger, Director of Information Resources
Temple University, Philadelphia, Pennsylvania 19122

Some colleges and universities have developed on-line systems for information retrieval from student or financial data bases. Others intend to develop similar systems in the near future.

For applications where a single discrete record is to be retrieved and on-line update is not permitted, microfiche systems are less expensive and more reliable than on-line systems. They can compete with on-line systems in retrieval speed and they can easily retain historical data when necessary.

1. THE BASIS FOR COMPARISON

The scale of individual systems can make a difference in the type of equipment used and the manner in which the data is handled; therefore it is necessary to specify some characteristics of a hypothetical system for purposes of comparison. Let us assume that we are developing a system whose primary function is data maintenance. This system will use either fifteen terminals or twelve microfiche viewers and three viewer printers to gain access to 30,000 student records. There will be thirty users making constant inquiries. Ten sets of duplicate microfiche will be required and occasional hard copies of student records will have to be made. An access speed of fifty seconds per record will be adequate. Access to a record will be through the use of a student's name.

2. COMPARISON OF ACCESS SPEED

Several factors will determine the time required to display a student record. Since the student's name has been selected as the file access key, both the computer system and the microfiche system would need an index to identify a unique record. For most computer systems response time to a command from a given terminal is inversely related to the number of terminals accessing the computer at that time. Furthermore, few people can afford the luxury of having a computer dedicated to on-line inquiry systems. Most computer systems do batch, remote batch and on-line jobs concurrently; the number and type of batch jobs will affect the response to on-line inquiry jobs. Since most hard copy computer terminals operate at a maximum speed of thirty characters per second, it would take several seconds to print a student's record after it has been located. CRT terminals would not suffer from such a handicap.

Microfiche systems present some interesting contrasts to on-line terminal systems. Unlike on-line systems, speed is unrelated to the number of viewers in use; it is dependent solely upon the operator. When a record is located, a full page is displayed at once and viewer/printers can make copies of a given record in approximately ten seconds.

Although a record may be located more quickly with an on-line system when computer demand is low, the user does not have full use of a record until the terminal has displayed it. Thus microfiche systems become more competitive with on-line systems as user demand increases and in many instances response time is very similar.

3. COMPARISON OF RELIABILITY

Computer systems suffer reliability problems from both hardware and software. Hardware failures occur despite preventive maintenance. Data may be destroyed during a hardware failure such as a disk "head crash" or improper head location during a write operation. The duration of a failure is inversely related to the complexity of the computer system. Operating systems malfunction as do user-oriented software systems. All of these problems require the intervention of trained technicians to correct them. When the computer is "down" data is inaccessible to everyone.

The simplicity of microfiche systems makes them much more reliable than on-line systems. Many microfiche viewers have fewer then ten moving parts, of which the least reliable part is the lamp which can easily be replaced by the operator. Viewer/printers are much more complex and less reliable than viewers; however, only a few failures require the intervention of a trained technician. The possibility of data loss due to hardware failure is negligible and at any rate duplicate fiche are available. There is no software involved with the use of the microfiche retrieval system. Most importantly, when one viewer malfunctions, all of the other viewers are unaffected and data access can continue, though at a slightly reduced rate. Graceful system degradation is a feature that few computer systems can offer.

4. COMPARISON OF COST

To support an on-line system substantial additional hardware must be provided. Modems, additional core storage and disk storage space are needed. On-line inquiry systems cause additional CPU overhead because of the need to manage data going to and from the terminals. Terminals rent for approximately $100 per month. An on-line system such as the one I have described could easily cost $2,000 per month.

A microfiche system can be operated at much lower cost. Microfiche viewers cost an average of $150; viewer/printers cost $1,500 plus $50 per month for maintenance. Film, including duplicate film, for the system I have described, might cost $2,000 per year if a reduction ratio of 48X is used. The net savings per year may be many thousands of dollars. Moreover, microfiche is a less expensive storage medium for historical data than computer magnetic tape.

5. SYMBIOSIS: A BETTER WAY

Up to this point we have been comparing and contrasting on-line retrieval systems to microfiche retrieval systems. It is clear that the features of these systems, when incorporated into one integrated system, complement each other. On-line systems can locate data quickly and microfiche systems can store data reliably and inexpensively. If an on-line system were designed to locate records in a microfiche system vast amounts of data could be stored inexpensively and accessed rapidly.

COMPUTER SYSTEM PERFORMANCE, MEASUREMENT, AND ANALYSIS

Dr. James C. Browne
Department of Computer Science
and the Computation Center
University of Texas

COMPUTER SYSTEM PERFORMANCE, MEASUREMENT, AND ANALYSIS

J. C. Browne
Department of Computer Science and the Computation Center
University of Texas
Austin, Texas 78712

This paper presents an overview of the motivation for and the definitional procedures for computer system performance measurement and analysis. It is frankly intended to motivate the reader to include performance measurement and analysis as a routine portion of computer system operation. The actual gathering of data is only a very small portion of a computer system performance measurement and analysis project. The most significant portion of the project is the development of a model of the system which is to be measured and the definition of the measurement and analysis processes.

1. INTRODUCTION

It is the purpose of this paper to provide at a fairly high level a rather concrete view of the motivation for carrying out computer performance evaluation studies and the definitional procedures and requirements for a successful computer performance evaluation study. Papers of similar intent but with different emphasis have been prepared by Bell, Boehm and Watson [1], by Browne [2] and by Estrin, Muntz and Uzgalis [3]. The content of this paper will primarily be organized around the type of questions that are typically asked by computer installations considering performance measurement and analysis projects. There are, of course, many forms for the questions. A representative set of questions may include: What is performance, measurement, and analysis? Do I need performance measurement? What do I measure? How do I measure? How do I evaluate data? How do I use the data to gain my goals? What are the requirements for a profitable computer performance evaluation project? What is the state of the art in computer performance measurement and analysis?

2. WHAT IS PERFORMANCE MEASUREMENT AND ANALYSIS?

Stated in simplest terms computer performance measurement and analysis is the application of the principle of accountability and the principles of system engineering to computer system usage, operation, and management. This definition suggests naturally that any computer installation of moderate size and complexity should have a performance measurement and analysis project as a part of its routine operational procedures. This is particularly true for the computer installations in institutional business management offices. Business management offices are most often responsible for developing accountability data for their legislatures, boards of trustees, and for the federal government. The business office itself should therefore be a model of application of the principle of accountability and should be very conscious of the need for system performance measurement and analysis. Figure 1 illustrates the process of computer system performance measurement and analysis.

Systems Analysis
↓
System Model Definition
↓
Data Gathering
↓
Data Analysis
↓
Model Evaluation and Predictive Analysis

Figure 1. The Process of Performance Measurement and Analysis

The first two steps of the process, systems analysis and system model definition and the fourth step of data analysis are the ones which we focus on most strongly in this article. The third and fourth steps, data gathering and data analysis defines the process that is commonly thought of as performance measurement and analysis. We discuss step three relatively little in this manuscript. The last step, model evaluation and predictive analysis, is a step that is almost universally neglected in most installations even those which are fairly sophisticated in performance measurement and analysis. It is by the use of model evaluation that the utility of the data gathered and analyzed can be extended to serve a variety of purposes and goals.

3. DO I NEED PERFORMANCE MEASUREMENT?

It is the thesis of this article that the principle of accountability demands that all sizeable computer installations carry out performance measurement and analysis in order to establish the level of utilization of their resources whether or not they are interested or required to utilize them efficiently or to justify their practices. There are, however, a number of specific problems to which the techniques of computer system measurement and analysis are traditionally applied. These include:

(i) The location of problem areas or bottlenecks in the system.

(ii) Analysis of resource requirements of present and future workloads.

(iii) Location of outright errors in system software.

(iv) Prediction of system behavior under stress conditions.

(v) Guide lines for purchase and/or system enhancement decisions.

(vi) Monitoring of system operations.

(vii) Construction and validation of system models.

I know of no computer system installation which does not need computer system performance measurement and analysis for at least one if not more of these reasons. In fact, I can say that I am not acquainted with a computer system of any size which should not be carrying out a continuing and coherent program of computer performance, measurement, and analysis.

4. WHAT DO I MEASURE?

The success or failure of a computer performance evaluation project is to a very high degree determined by the thoroughness and knowledgeability which goes into the definitional phases of the measurement and analysis process. It is that aspect on which we shall focus to answer this question of what do I measure.

4.1 Definition of the All-over Measurement Process. The first step in any systems engineering project is to determine what are the goals of the project. Typically goals for performance measurements projects are: reporting for accountability, analysis to obtain increased effectiveness from the current system, or any other of the goals which are listed in section 3. The goals define not only the data requirements of the project but the techniques for analysis and the depth to which the analysis must be pursued.

The key element in defining the data required is to define a model of the system. It is normally the case that the total computer system in a large scale data processing installation is entirely too complex to serve as a framework for analysis. It is necessary, therefore, to abstract from the system those elements which are either or primary interest or which are the rate determining steps or devices in the system which is being monitored or measured. The system model can be as simple as merely a model of the central processor as a "black box." The model may include channels or the disk system, or it may be a much more complex model tending toward an abstracted realization of total system processing. The model may be a formal definition in terms of diagrams, mathematical equations, etc.; or it may be a verbal description which defines the devices and processes upon which interest is to be focused. However, an explicit statement of the model system is of utmost importance.

When a model has been defined, detailing of the model produces the three primary categories of data which must be developed. The first category is that of the definition of the workload which is to be measured. The second category is the resource utilization of the workload and the third category are the metrics which are to be used to evaluate performance since these must also be measured.

4.2 Workload Definition and Measurement. There are a number of basic questions which need to be answered with respect to the definition of the system workload. Is it to be taken in a production environment with the resulting instabilities and variations in workload? Or should it be taken in an artificial environment which is constructed to be representative of the real problem areas, but which has the property of reproducibility so that the system work load properties can be characterized and understood and changes in the system evaluated in a constant environment?

In most cases measurements are taken and the workload defined by working in the actual production environment. I feel, however, that the use of synthetic job mixes and sets of benchmarks to define environments for performance evaluation studies will increase as the sophistication of the systems engineering techniques being applied increases.

Another problem which must be answered is the level at which the workload is to be defined. Is it to be defined at the job or task level, such as FORTRAN compile, COBOL compile, user program executions? Is it to be defined at the job control language level in which each control card or job step defines a portion of the workload? More detailed specification such as the logical resource requirements as defined by the operating system are possible. The most common workload measure is the physical device utilization requirements of the workload.

4.3 Resource Definition and Measurement. There are two important elements in the definition of resource measurement which need to be further discussed. The first of these is the level at which the resource utilization is defined. It may be defined at the process level i.e., a certain COBOL program or payroll process utilizes CPU service at a rate of so many seconds per hour of oepration. Alternatively, the resource utilization may be cast at the system level. At this level resources are primarily accounted for in terms of the resources consumed by processes which are defined by the system. A disk access requires a certain amount of CPU services, a certain amount of channel service, and certain memory buffers. Finally, the most common definition is to measure resources at the level of the devices in the system i.e., a channel is used a certain fraction of the time, the CPU is busy a certain fraction of the time, and is engaged in the monitor functions a certain fraction of the time, etc.

The other primary issue in resource utilization measurement is the retention of correlations both within a level and between levels. Correlations may be obtained by direct measurement or by inference. An example would be the sequence of activities such as a CPU burst followed channel reservation, disk track reservation, disk unit reservation, etc.

4.4 Measurement and Definition of Metrics. The final set of data which must be defined are performance metrics. It is ordinarily the case that the metrics of interest are simple ones such as throughput, response time, CPU utilization, or other such relatively simple measures. The real questions dealing with definition and measurement of metrics are in being certain that one has made adequate provision in the measurement process for the validation of the system model and for determining the sensitivity of the metric to the set of variables which may occur in the measurement process either purposely or because of normal fluctuations in workload. An example might be how sensitive is CPU utilization to the queue length on the input queue. One would be foolish to try to assess the efficiency of a CPU scheduler without monitoring the job queue. Another factor is to be certain that one can, with a set of metrics defined and measured, resolve the effects on the metrics of the different factors of hardware, software, and workload characteristics.

5. HOW DO I MEASURE (COLLECT DATA)?

It is a fact that nearly all systems collect data suitable for performance measurement and analysis under the guise of system accounting. Most of the modern multi-programmed operating systems provided by major vendors have extremely elaborate accounting systems. Indeed these accounting systems are very often expandable at the user option to obtain a variety of data. It is thus the case that an accounting system can serve as the basis for a very large fraction of all the data required for performance measurement and analysis. The primary problem which remains here is the analysis and formatting of the data.

Hardware monitors are another standard set of devices for data gathering. Hardware monitors have been refined to a fairly precise state. A number of vendors have available hardware monitors which they will lease or sell or which they will bring into your installation and attach to your system. The hardware monitors have well defined probe points for most major computer systems. They are supported in most cases by fairly adequate data analysis procedures. The basic problem with hardware monitors is that they do not establish correlations between executing processes and the utilization of resources. An additional limitation is that they normally are able to resolve resource utilization only at the level of individually identifiable physical devices and not in more complex or logical units.

Software monitors are much more flexible and versatile than hardware monitors. Software monitors for IBM equipment both at the operating system level and at the application program level are available from several vendors. These products have a history of paying a high return on their utilization.

There is, of course, the possibility of an installation developing its own software monitors. It is a part of the folklore that such "homemade" software monitors are difficult to write and validate and tend to destroy the integrity of the system as well as consume large amounts of system resources. Our experiences with installation tailored software measure products has, however, been very satisfactory.

6. HOW DO I EVALUATE DATA?

The analysis and definition process leading up to the measurement of data will normally suggest the procedures and requirements for evaluating the data after it is measured. It is usually the case that simple cumulative and statistical analyses are sufficient to provide a basis for operational reporting and for accountability reporting. More sophisticated analysis may be required for evaluating changes in performance caused by variations in hardware or software configuration.

It is usually the case that data reduction and analysis packages are available for accounting system data and for the standard hardware and software monitor data. These may or may not meet the requirements of a particular installation. It is usually the case that the data obtained from these sources is sufficiently well formatted so that analysis programs are fairly readily written.

There are, however, a number of problems still present in the analysis of data. Standard norms of acceptable performance levels for devices and resource utilization by processes are for the most part not yet widely available. They are certainly not available in a form which facilitates comparison between different types of systems and environments. Another somewhat confusing factor is that the ranges of effectiveness for device utilization tend to vary quite substantially. For example a CPU utilization of 40% would be regarded as not very satisfactory while a utilization of 40% for a disk unit would be regarded as a heavy utilization.

An effective means of data analysis is to use it to parameterize a formal or mathematical model of the system. This useage of the data often provides a basis for extending the range of the measured data into analyses of the potential performance of unmeasured or unrealized system configurations.

7. THE REQUIREMENTS FOR A PROFITABLE COMPUTER PERFORMANCE MEASUREMENT AND ANALYSIS PROGRAM

The first and most important element of a successful performance evaluation project is a comprehension and understanding of the goals of the institution which the computer installation serves and the purposes of the computer installation. Without this there can be no communication or effective long-range utilization of the results of a performance measurement and analysis project.

The next important element is a thorough knowledge of system characteristics. The hardware system, the operating system, the data management system, the COBOL compiler, etc., all must be well understood if the data is to be viewed in proper context. A knowledge of the workload characteristics is also essential. Finally, an operational knowledge of the techniques of systems analysis and systems modeling is required.

8. THE ROLE OF MODELING

We have repeatedly discussed the use of models in determining what data is to be gathered and in providing a basis for the types of analysis which are to be carried out on data after it has been obtained. The more formal uses of modeling can also be very rewarding. The construction of a mathematical representation of the system, i.e., a model, can be used to greatly extend the range of applicability of a set of data which have been measured and analyzed. The model of a system can be used to explore a large space of configurations, can be used to assess the role of unrealized systems or as yet non-available components within an installation. A variety of workloads can be processed on the model system and their impact upon the system assessed. A model can be used to return a variety of measures which may be difficult or expensive to measure.

There are, of course, many difficulties with the modeling procedure. The tools, techniques and expertise to apply modeling in a satisfactory and rewarding way are not generally available. Models which are realistic enough to return information with a high degree of credibility are usually cumbersome to evaluate. Finally, there is the possibility that conceptual errors may creep into the model. I should point out however that the most important role of modeling is often that of forcing an understanding of the system before measurement takes place.

9. MEASUREMENT--STATE-OF-THE-ART

We summarize here for convenience our viewpoint of the state-of-the-art in computer system performance measurement and analysis. It is the case now that device utilizations, i.e., resource useage at the device level, can be very effectively resolved. It is also the case that external rates of workload processing such as thruput, response time, etc., can usually be readily and effectively measured. It is seldom the case, however, that measurements can directly return functional relationships. Quite elaborate measurement and analysis techniques are usually required to carry out this function.

It is seldom the case that we are able to measure the absolute capacity of a system for doing work.

In order to make progress toward becoming a more easily applicable and widely available procedure, computer performance measurement and analysis needs better theory and models. Experiments need to be carried out in controlled environments so that we can establish definition of some standard workloads and definitions of standard values of metrics for these workloads.

REFERENCES

[1] T. E. Bell, B. W. Boehm and R. A. Watson, Framework, and Initial Phases for Computer Performance Improvement, Proceedings of the 1972 Fall Joint Computer Conference. vol. 36, 1972, 1141-1154.

[2] J. C. Browne, Functional Concepts and Basic Data of Performance Measurement and Evaluation, Proceedings of the First Texas Conference on Computer Systems, vol. 1, 1972, 1-5.

[3] G. Estrin, R. R. Muntz and R. C. Uzgalis, Modeling; Measurement and Computer Power, Proceedings of the 1972 Spring Joint Computer Conference, vol. 35, 1972, 725-738.

OFFICE AUTOMATION AND HYBRID INFORMATION SYSTEMS

Dr. Jack E. Shemer
Principal Scientist & Area Manager
Advanced Development Laboratory
Xerox Corporation

OFFICE AUTOMATION AND HYBRID INFORMATION SYSTEMS

Dr. Jack E. Shemer
Principal Scientist & Area Manager
Advanced Development Laboratory
Xerox Corporation

Information is primarily perceived via the audio and visual senses. In the office environment, there is a tremendous dependence upon conveying information in image form. Each of us is well aware of the countless reports, financial summaries, business forms, letters, etc., that cross our desk each day.

Information which is presented in image form may be classified on the basis of its storage state in the database from which it is produced. In this context, there are three classification states: 1) human readable, 2) machine readable, and 3) machine readable/machine processable. Human readable information is exemplified by an image on paper or film with the fundamental information element being an alphanumeric character, a mark or bounded contour. Information stored in a machine readable state is typified by electrical or optical storage of analog or digital information (e.g., video tape) which can be translated by a machine to generate an image, where the fundamental information element is a picture element (pixel), and there is a one-to-one correspondence between the image and an identified storage record. Machine readable/machine processable information is characterized by the electrical or optical store of digital or analog signals that can be read, translated, processed and interpreted to generate one or more human readable image forms, where the fundamental information element is an encoding of an alphanumeric character, a stroke, or bounded contour.

Prior to 1950, essentially 100% of the administrative database was stored in human readable/graphical form, principally as images recorded on paper. With the advent of computer technology and Business Data Processing (BDP) applications, a dramatic shift occurred. Today, major corporations depend upon information services performed on a database which is stored in machine readable/machine processable form on digital recording media. The capture of information in this form has made possible a great variety of automated services.

In addition to the growth of BDP services, the advancement of computer and communication technologies has had a pronounced effect on the emphasis of communication. In 1950, information was communicated primarily on the basis of a need to know. In 1975, information communication is still sender controlled, but now the question is, "Will it hurt to tell?" By 1990, information conveyance will be receipient controlled. Rather, the question will be, "Do I want to know?" Accompanying this evolution will be the continuing dramatic information explosion and a shift in the database from human readable form to mixed forms of storage as depicted in Figure 1.

As new information store technologies emerge, a fundamental challenge will be to provide information services upon databases stored on a variety of media in graphical/human readable form, machine readable form, or machine readable/machine processable form. In the office environment, the challenge is to provide a new variety of solutions to the classical problems of creation, conversion, store, transform, retrieval, manipulate, process, and distribute. For these, a fundamental theme will be timeliness, i.e., responsive retrieval and distribution of information to the right place and in the right format. To achieve this, we can expect to witness the emergence of new devices ranging from "communicating marking engines" to "electronic file" cabinets. A sample of these elements is presented in Figure 2.

The hardware requirements for such systems dictate that they be synthesized from a broad variety of technologies. Processors will be heterogeneous in form and designed to autonomously perform specific functions. Advancements in storage technologies will yield hybrid systems derived from semiconductor, magnetic, and optical storage devices. Figure 3 provides an overview of the expected elements that will be required. The attendant system architecture for these administrative systems of the future will be heavily biased by considerations of stand-alone capability, functional extensibility and communications. Hence, particular emphasis will be given to distributed function architecture and the interconnection of heterogeneous subsystems.

FIGURE 1

THE EVOLUTION OF COMPUTER AND COMMUNICATION TECHNOLOGIES HAS CAUSED A DRAMATIC SHIFT IN THE DATABASE

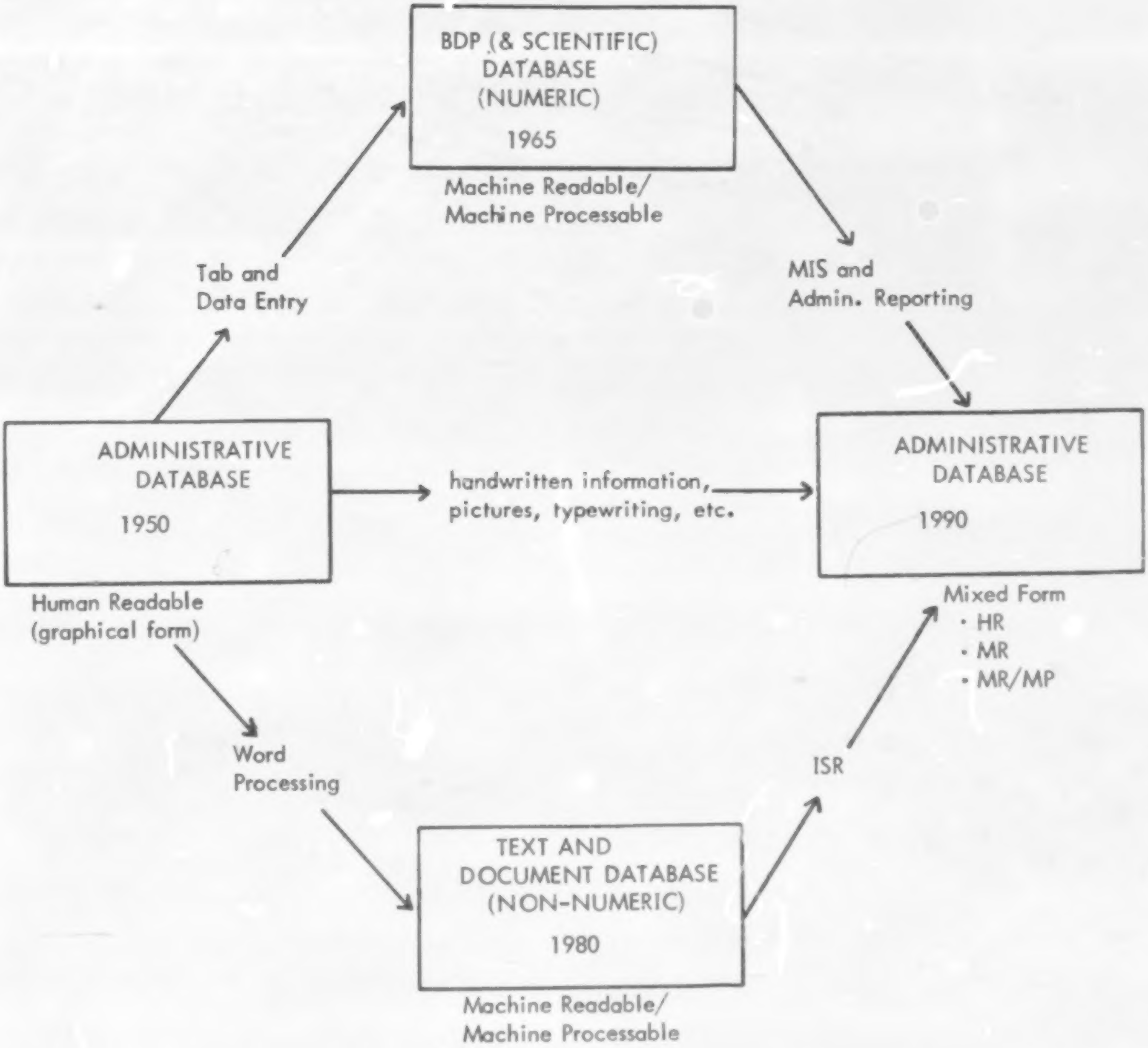

BDP (& SCIENTIFIC) DATABASE (NUMERIC)
1965
Machine Readable/ Machine Processable
Tab and Data Entry
MIS and Admin. Reporting
ADMINISTRATIVE DATABASE
1950
Human Readable (graphical form)
handwritten information, pictures, typewriting, etc.
ADMINISTRATIVE DATABASE
1990
Mixed Form
· HR
· MR
· MR/MP
Word Processing
ISR
TEXT AND DOCUMENT DATABASE (NON-NUMERIC)
1980
Machine Readable/ Machine Processable

FIGURE 2

WHAT TYPES OF DEVICES MIGHT WE EXPECT TO SEE IN TOMORROW'S OFFICE?

- Communicating Marking Engines (e.g., FACSIMILE elements interconnected locally and remotely)
- Document Creation Devices (word processors with local files)
- Personalized Electronic Index (electronic disk, keyboard, processor, and alphanumeric output employed to locate information in one's personal file)
- Office Work Station (keyboard, display, removable store, hard copy output, audio, and communications with one version for a secretary and another for an administrator)
- Large Database Stores - mixed forms of information forming an "electronic file"
- Demountable and Portable Store
- Image Recognition and Decoding Elements (OCR, graphic encoders, etc.)
- High Speed Hard Copy with Typographic Quality
- Distributed Networks of Such Elements (with capability to connect to larger computer systems)

FIGURE 3

HARDWARE REQUIREMENTS DICTATE HYBRID INFORMATION SYSTEMS

Input/Output:

- Keyboards (QWERTY, program function, etc.)
- Character printers (various speeds and image quality)
- Display (resolution 2K X 2K, high contrast, bright, rapid erase, low flicker, also desirable to have color and image capture, store, and readout capability)
- Cursor positioning device
- Tablet for graphic input (ideally on flat panel display)
- High speed marking engines (medium to high quality, fast copy, 1/sec. on-demand via non-impact printing technology)

Processors ("styled for function"):

- Image encoders/decoders (digital filtering)
- Text manipulators
- Display and print processors - a) alphanumeric, b) graphics, and c) composite
- Communication control (local/remote)
- Hierarchical memory control
- Associative processors (high speed serial search and interpretation for ISR)

Storage (semiconductor, magnetic, and optical stores)

- INDEX $\approx 10^6$ characters per user stored in digital form, readable at 2 Mbyte/sec.
- IMAGE $\approx 10^4$ pages per user stored in MR or MR/MP form with random access ≈ 2 sec.
- ARCHIVE $\approx 10^7$ pages per department stored as micro-images, video, or electronic facsimile which can be viewed or output at random with ≈ 10s access, with sequential access of 1/sec. after locating first in a list.

Communication Nets:

- Local, broadband, multichannel 5MHz/channel
- Remote common carrier
- Disciplines and protocol dependent upon the form in which the information is stored, transmission medium, and network topology.

TRACK V

Vendors

Track Coordinator: Richard L. Mann

The Vendor Track of the CAUSE National Conference offers presentations of products of interest to the higher education data processing community. These sessions feature professional presentations rather than exhibits or displays, and the use of products in colleges and universities is emphasized. An abstract of each presentation is included and more information may be obtained by contacting representatives of the vendor. This year's vendor participants include:

Azrex, Inc.
CINCOM Systems, Inc.
Cullinane Corporation
Digital Equipment Corporation
IBM Corporation
Informatics
Information Associates, Inc.
Integral Systems
MRI Systems Corporation
Pansophic Systems, Inc.
Performance Development Corporation
Software AG
Systems & Computer Technology Co.
Wachovia Services Inc.
3M Company

STUDENT LOAN SYSTEMS

Dan R. Ackley
Institutional Marketing Manager
Wachovia Serices, Incorporated

WACHOVIA SERVICES, INC.

Wachovia Services, Inc. has an automated student loan system for the lender who must maintain its own in-house billing and bookkeeping system.

The system is the product of ten year's involvement in student loan processing. In 1964 we began work on our first NDSL system: in 1968 we modified that system to accommodate the latest changes in loan legislation. In 1973 we redesigned the system from the ground up, and it is now the most complete and sophisticated National Defense/Direct Education Act Loan processing system available. We now use this system to process more than 350,000 NDSL, NTA, HPA, and Institutional Loans for 450 lenders ...lenders with as few as 50 loans or as many as 15,000.

Some facts about our system:

- All programs are written in ANS/COBOL.
- Only one master file is used.
- All files are arranged sequentially.
- There are five programs included in the system:
    1) Monthly Update
    2) Reports (Both weekly and monthly)
    3) Weekly Post
    4) Weekly Input Edit
    5) Header Update

Wachovia will be able to provide you with a complete, integrated, student loan billing system. This program will aid you in the following ways:

- Administration of the total program.
- Reduce the total manpower required to support the student aid office.
- Provide management reporting previously unavailable.
- Stabilize or even reduce the in-house costs associated with the student aid program.

If you would like additional information concerning the system that we have designed for the student loan programs, please call or write to us at:

Wachovia Services, Inc.
Institutional Marketing Department
Post Office Box 3176
Winston-Salem, North Carolina 27102
Phone: 919/748-5608

S73

A Member Company of The Wachovia Corporation

IDMS - INTEGRATED DATABASE MANAGEMENT SYSTEM

CULPRIT - INFORMATION RETRIEVAL SYSTEM

William Casey
Technical Representative
Cullinane Corporation

IDMS - INTEGRATED DATABASE MANAGEMENT SYSTEM

CULPRIT - INFORMATION RETRIEVAL SYSTEM

William Casey
Technical Representative
Cullinane Corporation

The Cullinane Corporation made technical presentations on two products: IDMS, Integrated Database Management System, which is the only system available for IBM hardware that meets the subset of the CODASYL Data Base Task Group's specifications; and CULPRIT, Information Retrieval System, with emphasis on those versions for use with Data Base Management Systems including IDMS, IMS, and TOTAL, as well as special user department versions such as EDP-AUDITOR and PERSONNEL-EDP-REPORTER.

MARK IV - FILE MANAGEMENT SYSTEM

Al Seyler
Sales Manager
MARK IV Systems Company

MARK IV - FILE MANAGEMENT SYSTEM

Al Seyler
Sales Manager
MARK IV Systems Company

Mr. Seyler explored the concepts of the data base as they relate to a file management system. The MARK IV File Management System was illustrated on how it interfaces with data bases in the college and university administrative systems environment.

SYSTEM 2000 AS AN ADMINISTRATIVE AND SCHOLASTIC AID

Charles J. Lewis
Data Management Consultant
MRI Systems Corporation

mri SYSTEMS CORPORATION
12575 RESEARCH BLVD AUSTIN TEXAS

ABSTRACT

"SYSTEM 2000 AS AN ADMINISTRATIVE AND SCHOLASTIC AID"

by

Charles J. Lewis
Data Management Consultant
MRI Systems Corporation
Austin, Texas

This paper describes the use of SYSTEM 2000, a generalized data management system, within the college - university environment.

It begins with a description of SYSTEM 2000's features and the advantages which can be derived from its use. Then it cites specific applications of the system within various colleges and universities.

Those applications include:

University of Quebec Research Center	- computer usage control, research (serves ten research centers), administrative uses.
University of Indiana	- student data base (still in design stage), data dictionary.
Florida Board of Regents Information System	- various administrative and academic activities.
Central Texas College	- various in-house administrative and educational requirements, service to Army.
Texas A & M University	- data base for Southern Association of Colleges and Schools, railroad crossings data base (for engineering purposes).
The University of Texas	- various research uses including bibliographic data bases.
The University of Minnesota	- administrative and research uses.

For further information about SYSTEM 2000, please contact Charles J. Lewis or Forrest F. Preece, MRI Systems Corporation, P. O. Box 9968, Austin, Texas 78766, (512) 258-5171.

TELEPHONE 512/258-5171 MAILING ADDRESS P.O. BOX 9968 AUSTIN TEXAS 78766 TWX 910 874 1316 CABLE MRI

Vol. I

705

ADABAS - ADAPTABLE DATA BASE SYSTEM

John N. Maguire
President
Software Ag

software ag

ADABAS (CAUSE Presentation by John Maguire, software ag of North America, Inc.)

An idealized data base management system should:

1. Take over all existing files and data and integrate them into the data base automatically without any re-programming.
2. Have no restrictions concerning structure, ordering, or any special data format.
3. Provide a free form retrieval capability so that individual records or subsets of the data base may be easily accessed.
4. Allow that information from different files may be coupled automatically.
5. Be able to operate in an on-line and batch processing mode simultaneously.
6. Provide an efficient checkpoint and restart capability as part of complete automatic data protection.
7. Provide a multi-level data security function.
8. Allow that each field may be expanded, or cancelled at any time.
9. Allow that the logical definition of any record may be expanded at any time, without any effect on existing programs.
10. Allow and guarantee that the data base may grow continuously without having to reorganize.
11. Provide random access capability so that retrieval speeds are almost wholly independent of data base volume.
12. Provide for sequential processing speeds comparable to stand alone applications.
13. Optimize search strategy by using knowledge of the system contents of the data base.
14. Handle automatically the administration of all data storage devices assigned to the system.
15. Be ' forgiving ' and flexible in retrieval so that records that may possibly satisfy search criteria will also be identified, (e.g. names that sound like SMITH, but spelled differently will be considered).
16. Provide a multiprogramming capability for simultaneous use of the single system by many programs.

ADABAS has successfully accomplished all of the previously described features, and many more, in diverse installations on byte-oriented machines including the IBM 360/370 series, while requiring only 150 K bytes of core .... without using overlays.

THE AQUINAS PRINCIPLE AND ITS APPLICATION TO THE IMPLEMENTATION OF DATA BASE PAYROLL PERSONNEL SYSTEMS

David A. Duffield
President
Integral Systems, Inc.

Michael P. Aherne
Vice President
Integral Systems, Inc.

integral systems, inc.
eight main street • flemington, new jersey 08822 • (201) 782-3600

THE AQUINAS PRINCIPLE* and its Application to the Implementation of Data Base Payroll/Personnel Systems

Integral Systems specializes in the development and installation of modern, operational administrative systems for Higher Education, with special emphasis on Payroll/Personnel Systems attuned to the unique needs of colleges and universities. The Aquinas Principle is particularly relevant to Payroll/Personnel Systems in Higher Education, since many institutions find their operational stability dependent upon antiquated dysfunctional systems. The Integral Systems paper discussed the causes of this problem and some critical points to be considered in addressing it.

Changes in the institutional environment over the past decade have combined to significantly reduce the effectiveness and supportability of existing payroll/personnel systems.

Taken together, these changes have resulted in an increased administrative work load and in an expanded number of employees to be serviced by each support staff member; in a similar fashion the typical payroll/personnel system has become over-burdened with hasty changes, special purpose programs and limited suitability for change.

The urgency of the need for an improved payroll/personnel system and the attendent new facilities and reduced costs, sometimes leads institutions to underestimate the size of the effort required. ISI has found that adherence to the following guidelines produces the most effective new system.

-- Recognize the real dimensions of the task:
-- Set comprehensive goals:
-- Use the tools available to avoid the need for miracles:

Integral Systems provides a modern, operational Payroll/Personnel System developed in and for Higher Education. The system utilizes parameter-driven and table-driven features to facilitate installation and is structured to permit changes both to simplify its use and to adapt to a changing institutional environment. The system is available in modules to fit the needs of the institution. The system has been installed at a number of institutions--both large and small as well as public and private.

For further information on the Integral Systems Payroll/Personnel System or on other services provided by Integral Systems please contact: Michael Aherne, Integral Systems, Inc., 8 Main Street, Flemington, New Jersey 08822 (201) 782-3600.

*Not only believe in miracles, rely on them!

HOW TO MAKE THE BEST USE OF CONSULTING COMPANIES

John Robinson
Chairman
Information Associates, Inc.

HOW TO MAKE THE BEST USE OF CONSULTING COMPANIES

John Robinson
Chairman
Information Associates, Inc.

This presentation focused primarily on how colleges and universities can make the best use of consulting companies in the areas of administrative systems development, implementation, and operations.

INQUIRY SYSTEMS - QUERY 5

David McIlhenny
President
AZREX, INCORPORATED

AZREX INC.

3 Mountain Road, Burlington, Mass. 01803
(617) 272-8750

INQUIRY SYSTEMS - QUERY 5

David McIlhenny
President
AZREX, INCORPORATED

This session featured a presentation of inquiry systems with a complete explanation of QUERY 5, a software produce of AZREX, Incorporated.

INTERACTIVE QUERY AND REPORT PROCESSOR (IQRP)

J. F. Movizzo
IBM Corporation

Interactive Query and Report Processor (IQRP)

Developed by: IBM Corporation
Corporate Information Systems
Harrison, New York

The Interactive Query and Report Processor (IQRP) enables the end user to interact, from a terminal, directly with files of interest to him using a free form set of commands which can be easy to learn. IBM experience with predecessor systems indicates that providing this type of query and reporting facility to the end user potentially constitutes a significant productivity enhancement and can permit a wide variety of applications to be implemented without traditional programming.

The key to IQRP's value in permitting the end user to generate interactively his own applications is a simple, flexible, non-procedural, powerful set of commands. Use of IQRP predecessor systems within IBM has demonstrated that clerks, managers administrative personnel, technical staff and production workers can quickly learn to use the IQRP query terms.

Features of IQRP

Highlights of the features of IQRP include the following

- *Easy to use.* The fundamental English-like simplicity of the IQRP instructions can make it easy for new users to learn IQRP. A user can define his own working terms in addition to existing IQRP terms using a language entirely familar, comfortable and mnemonic to him. Such terms can define and/or represent a word, a phase, or an entire report. report.
- *Interactive Communication.* A user's request for IQRP service can be checked interactively for completeness and correctness. The user can see the result of his request at the terminal. This permits him to modify and resubmit additional requests at once if he is not satisfied with the results. This technique allows him to pursue an investigation prompted by the results from his first request, as, for example, by making new requests suggested from previous results.
- *Data and File Security.* Password protection at signon time is provided. IQRP provides a user profile that determines which files, records and fields within records an individual user may access. Another facility is conditional field security which is based upon specific contents of one or more fields.
- *Selective Logic.* Full Boolean logic is available to permit pinpointing the precise data to be used for a report. The contents of the fields to be examined may be compared with user-specified terms, the results of a calculation, or the contents of one or more other fields.
- *Extensive Sequencing Capability.* Data meeting selection criteria can be sequenced in either ascending or descending order. It is possible to have some data sorted in ascending order and some in descending order within the same report. IQRP also permits terminology such as top ten, low five, etc. IQRP further permits the user to request that totals and/or subtotals at various break-points be computed. Totals and/or subtotals may be displayed either with or without associated detailed data.
- *Arithmetic Capability.* Arithmetic operations can be invoked from a terminal, including exponentiation and square root. These operations can be applied to field contents as well as to subtotals and totals, or to all three. The results can be used in the selection of data for inclusion in a report.
- *Alternate Output Capability.* The user can route his output to a line printer for inquiries or report requests which generate voluminous output. Inquiry or report requests which are to be run in a batch mode can optionally be checked for syntax errors during a terminal session.
- *Implementation.* Repeated experience within IBM indicates that if the required control systems and files are already installed, an experienced programmer can usually learn IQRP, install it, define the desired files and authorized users, establish security requirements, teach the instructions to the users, and have meaningful applications installed in less than one month. Subsequent files can be easily added to the system, as required.

Benefits of IQRP

- Provides the ability to extend application development to non-programmers
- New applications may be easily installed
- Can minimize the need for traditional maintenance of end user applications by data processing professionals
- Can support the business decision makers' need for information
- Potentially enhances the utilization of existing and planned data resources
- Use can complement current application development

IQRP is an Installed User Program, distributed by IBM on an "as is" basis. Additional information, including systems considerations, can be found in the Availability Notice, G320-1544. Contact your local IBM representative for further details.

MICROFILM AND THE COMPUTER: BEST OF BOTH WORLDS

James Rasnick
National Coordinator of Education
3M Company

GENERAL OFFICES • 3M CENTER • SAINT PAUL, MINNESOTA 55101 • TEL. (612) 733-1110

Microfilm Products Division

MICROFILM AND THE COMPUTER: BEST OF BOTH WORLDS

James Rasnick
National Coordinator of Education
3M Company

This session featured the important uses of Computer Output Microfilm (COM) in college and university administrative applications.

HOW TO BE A HERO EVERY DAY OF DP LIFE

Robert Briggs
National Sales Manager
Pansophic Systems, Incorporated

HOW TO BE A HERO EVERY DAY OF DP LIFE

Pansophic Systems Incorporated

EXECUTIVE OFFICE
1301 WEST 22nd STREET
OAK BROOK, ILLINOIS 60521
(312) 325-9600
TWX 25-3299

Efficiency is the key to valuable data processing, for if data is handled accurately and quickly and information may be obtained readily and hardware can be utilized at its fullest capacity, then you, as DP management, are doing your job. With utility software from Pansophic Systems Incorporated, we hope to make that job easier with an ultimate goal of making you a hero in your computing environment.

PANVALET, The Program Management and Security System, provides you centralized storage of 80-column card images, whether they be source programs, object decks, data or JCL... all on disk in a compressed format. Data from more than 4 million cards stored on a single 3330 disk pack is readily available for maintenance and manipulation by programmers. PANVALET is designed to allow programmers the tools they need to easily write and maintain programs while at the same time, providing management with a complete audit trail of all library activity... all of this occurring with flawless PANVALET security in three phases: first, PANVALET provides total, current library backup to prevent data disaster. Second, 4 levels of PANVALET access codes allow management to make security on a specific member as tight as they wish. And third, PANVALET production programs can never be changed to TEST status, thus preventing accidental or intentional program alteration. PANVALET interfaces with CICS and CMS.

EASYTRIEVE, The Information Retrieval and File Management System, is the most popular report generation system in the world today because it is fast, easy to use and versatile. In colleges and universities, EASYTRIEVE has shown itself to be an indispensable tool for financial aid offices, for SEMIS reporting and for research projects. EASYTRIEVE code is English and automatically formats reports, extracting only the information from records that you specify. Do file management, one-time report requests, mailing labels and a thousand other jobs with EASYTRIEVE in moments of coding instead of days. And now, EASYTRIEVE users are provided with job accounting applications that produce reports from SMF data, and GRASP-like reports for DOS and DOS/VS. EASYTRIEVE now interfaces with IMS, TOTAL, ADABAS, etc.

PAN*DA, The Disk Analysis and Data Set Management System insures effective utilization of expensive disk space through user-group, volume and track map reports, listing more than 40 data sets per page and summaries at the end of every report. PAN*DA automatically flags potential problem areas, obsolete data sets or wasted space, allowing you to see trouble spots before they occur. PAN*DA user exits automatically do billing for disk space, PDS compression and automatic scratching of obsolete data sets.

Typically, Pansophic software cost-justifies itself in a shop in less than six months, providing pure savings from then on. More than 150 colleges and universities in the U.S. have discovered the efficiencies of these packages; there are more than 2,000 Pansophic installations around the world. It has been our pleasure to enlighten you about our truly marvelous data processing tools today and hope that you will make us prove to you that we can save money and time and increase your productivity with our products.

Robert S. Briggs
General Sales Manager

IMPLEMENTING SCT'S NON-PACKAGE LOCUS WITHIN BOTH BATCH AND DBMS ENVIRONMENTS

Jack F. Magosin
Vice President

John Baillis
Manager
Midwest Technical
Operations

Systems & Computer Technology

SYSTEMS & COMPUTER TECHNOLOGY CORPORATION

SEVEN NORTH FIVE POINT ROAD ■ WEST CHESTER, PENNSYLVANIA 19380 ■ 215-692-7990

ABSTRACT

"IMPLEMENTING SCT'S NON-PACKAGE LOCUS WITHIN BOTH BATCH AND DBMS ENVIRONMENTS"

This session addressed SCT's modus operandi of defining and implementing integrated student and financial information systems within both batch and DBMS environments. Over the past seven years SCT has been contracted by more than eighty colleges and universities to define, implement and provide training in computer based administrative systems. The results of these efforts are what SCT calls our non-package Library Of College and University Systems (LOCUS).

In order to illustrate how processing differs, an example of the student registration portion of an integrated student information system was discussed under both a sequential and DBMS environment.

Other topics addressed in the session included:

What is a data base management system?

Why do computer installations consider utilizing a DBMS?

What are the general advantages and disadvantages of a DBMS?

What DBMS's are available?

For additional information concerning SCT's systems consulting, systems development, proprietary systems, and implementation services, please contact:

Mr. Jack F. Magosin, Jr.
Vice President
Systems & Computer Technology Corporation (SCT)
West Chester, Pennsylvania 19380
215-692-7990

TOTAL - A SUCCESSFUL APPROACH TO DATA BASE REQUIREMENTS IN A UNIVERSITY ENVIRONMENT

Michael J. Ehrensberger
CINCOM Systems, Inc.

Cincom Systems, Inc.
2300 Montana Avenue, Cincinnati, Ohio 45211 Tel. 513 662-2300 TWX 810 461 2732

Cincom Systems develops and markets proprietary software. This software includes TOTAL, a data base management system, currently installed at over 800 locations, Environ/1, an on-line task monitor system, installed at over 150 locations, and Socrates, an information retrieval and display system. TOTAL is currently used extensively by colleges and universities throughout the United States, Canada and Europe. Applications implemented using TOTAL include student registration, student transcripts, payroll/personnel, library and administrative systems.

TOTAL is fast becoming the defacto standard for data base management systems. It is currently operational on the IBM System/3, IBM 360-370, NCR Century, Univac 9000 series, CDC Cyber series, Univac Spectra series, and the Honeywell OS 200-2000 series.

TOTAL has for the second year been named to the DATAPRO Honor Roll. As DATAPRO states:

> "The primary reasons why most of the users selected TOTAL are predictable: it's low on core, relatively easy to use and implement, flexible, and has a fairly high level of performance. Several of the users wanted a data base management system up and running 'fast.' And TOTAL was the fastest in giving them what they wanted. Also, some users indicated that they wanted to implement their applications gradually, adding new applications as they went along. And TOTAL does provide that flexibility also."
>
> "When DATAPRO 70 contacted the present users of Environ/1, however, it was with some astonishment that the following statements were received: Environ/1 'is a data processing dream come true' -- 'most sophisticated commercial system available' -- 'system is golden.'"

For additional information concerning TOTAL as well as the names of colleges, universities and secondary school systems using TOTAL, please contact Mr. Walter Muir at 513/662-2300.

MJE/el

ACHIEVING A PASSING GRADE IN DATA BASE

Leo J. Cohen
President
Performance Development Corporation

PERFORMANCE DEVELOPMENT CORPORATION

- IN COMPUTING
- IN SYSTEMS
- IN EDUCATION

32 SCOTCH ROAD • TRENTON, N. J. 08628 • (609) 883-3707

THE CORPORATE DATA BASE

An appreciation of the meaning of data base and the data base system, as opposed to the function or file oriented system, is essential to understanding the commitment that the data base decision implies. This commitment is extensive and involves relatively high risk, utilization of resources, and extended time. This is particularly true for the small system user in that the degree of commercial DBMS support currently available is rather thin. However, the data base technology offers many significant potential advantages for data processing in any context, and the development of a tailored data management capability is well within the capability of any data processing organization with a reasonably good track record.

This presentation will review the several aspects of the development of a corporate data base project. Seven primary steps are involved in the development of such a project, and they incorporate many of the decision making questions regarding a data base that must be answered. In addition, the management issues for successful operation of the data base system environment will be investigated under the heading of data base administration.

WISE - ADMINISTRATIVE DATA BASE MANAGEMENT ALONG WITH INSTRUCTIONAL TIME-SHARING

Robert H. Meese
Marketing Manager
Four-year Colleges
Digital Equipment Corporation

digital

"WISE: Administrative Data Base Management along with Instructional Timesharing"

* * ABSTRACT * *

WISE (the Wheaton Information System for Education), a proprietary software product recently developed and announced by Digital, is discussed as both a generalized on-line data base management system and a set of application programs suitable for college/university administrative data processing. The major features and benefits of the package are presented, which include outstanding adaptability and expandability, on-line orientation, ease of use, file security, capability for simultaneous instructional timesharing, and economy (complete running systems available for $85,000 to $200,000 purchase). The system's internal structure is explored, with emphasis on the DBMS features. Finally, a look at the system through the eyes of the terminal operators in various administrative departments is presented.

DIGITAL EQUIPMENT CORPORATION, 146 MAIN STREET, MAYNARD, MASSACHUSETTS 01754
(617)897-5111 TWX: 710-347-0212 TELEX: 94-8457